1973

Aquatic Models

Wayne Minshall

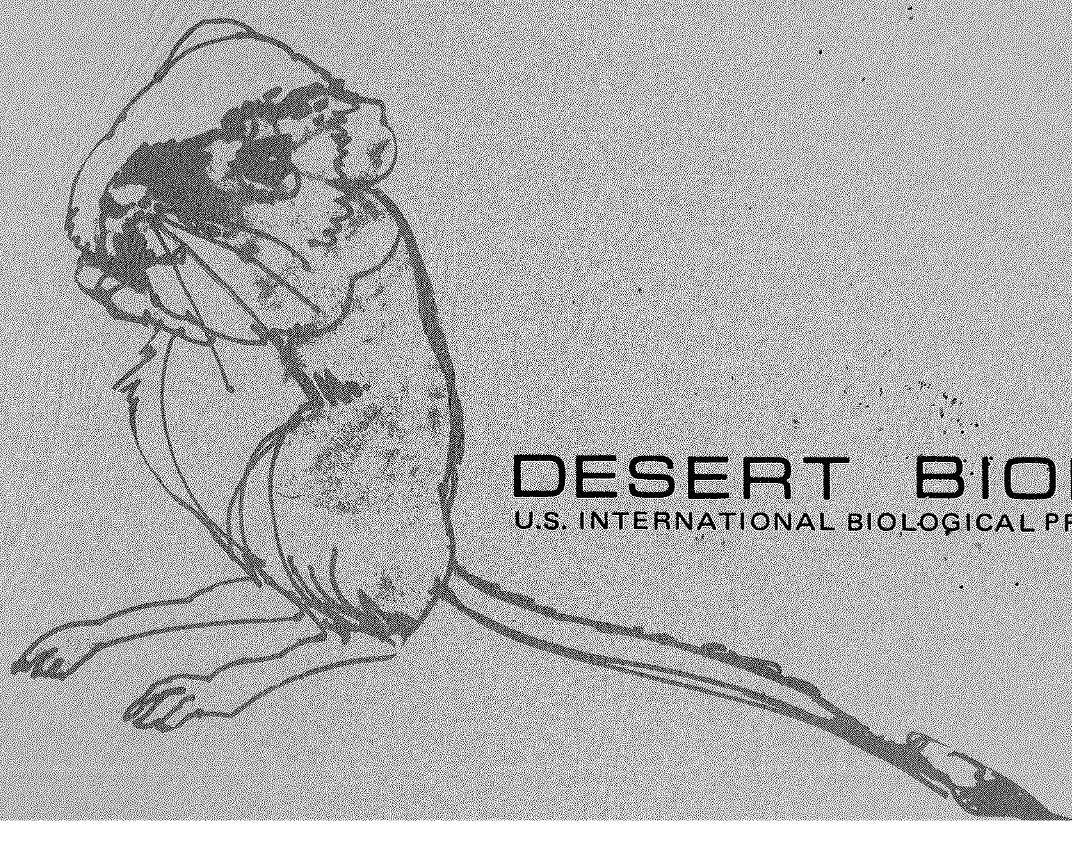
IBP

RESEARCH MEMORANDUM

RM 73-57

AQUATIC MODELS

Coordinator: Wayne Minshall
Idaho State University



DESERT BIOME

U.S. INTERNATIONAL BIOLOGICAL PROGRAM

II

AQUATIC MODELS

INTRODUCTION

David W. Goodall

These models are intended to predict the values of a wide range of variables in the system simultaneously. They may be expected to lack the precision possible in a model intended for a special purpose; but by easy modification (module replacement -- see below) they can serve special purposes reasonably well. And -- their prime function of the present state -- the fact that all elements in the system are considered simultaneously makes it possible to do a variety of tests to which special-purpose models would not lend themselves, and thus they may serve as a guide to research priorities.

The ecosystem is envisaged as horizontally homogeneous, consisting of a vertical prism of water, with the sediments below it, which may receive inputs from an adjacent area and provide output to another adjacent area. The model can thus represent either stationary or flowing water but in its present development does not cover horizontal heterogeneity, or processes in a horizontal direction except in terms of input and output.

Modular Structure

The models can be conceived as composed of a set of modules which act as "black boxes" to one another. For each, the inputs and outputs are determined by the general characteristics of the system, so that it operates on inputs and provides outputs which are part of the common "language". Internally, each of the modules or submodels may have a great variety of structure but this is a matter of indifference to the rest of the model, to which it is simply a "black box".

The modules are concerned with the various component parts, or sets of processes in the system. Initially, these are the plants; the animals; and the abiotic environment in which they exist. These modules or submodels may be further subdivided as required. Each submodel may be developed at different levels of detail, complexity, and sophistication, and these alternative submodels can then be combined freely to give models which may be highly sophisticated in respect of some components, simple in others.

The Variables

Wherever possible, all variables are expressed per unit area of water surface. The main state variables modelled are the quantities (per unit area) of different chemical constituents in different components of the biomass, which may be divided in a great variety of ways; in the components of detritus and other dead material (which may also be sub-divided in various ways); and in the water.

Population data and their changes may also be included, but as real variables representing an average per unit area rather than integral variables for a specific delimited region.

Exogenous variables include the dates and daily quantities of precipitation events, with the average composition of the precipitation received; the average daily amount of solid material falling into the water from above, month by month, with its type and composition; the dates of events where overland flow enters the system, with the amount of water, solutes and detritus of various types carried; the dates and amounts of water withdrawal (e.g., for irrigation); and mean monthly figures for water temperature, radiation intensity at the water surface, photoperiod, and pan evaporation rate. This list will be subject to modification as the model develops further.

Classification of Variables

In any ecosystem, the variables characterizing it may be classified in different ways for different purposes. Apart from the biological classification itself, one may classify different parts of the system as organ types, by age or stage of development, or to their topographical location, or one may classify variables according to whether they are quantities of chemical elements or population data, say. Plant and animal species may be classified not only according to their taxonomic position, but by life form, feeding habits, life history, etc. And all these different cross-classifications may be relevent to some part of the functioning of the ecosystem -- some of the processes leading to changes in the values of the state variables.

Cross-classification of state variables (and of some of the parameters of the system) is accordingly a dominant feature of the models. This makes it unnecessary to describe separately the processes in which each state variable is involved. Instead, it is necessary to give separate specifications only for those classifications and classes which are relevent to the particular process in question, all other classificatory sub-divisions of the state variables being ignored for this purpose. If, for instance, for a particular animal species, feeding habits are the same for mature and immature individuals, then the model uses a common description for feeding processes of the different age categories, while in that part of the model describing reproductive behavior it is clearly of first importnace to distinguish these age categories.

Functional Forms

Most of the variables being real, and most changes being continuous, it is usually appropriate to describe the rates of change in terms of differential equations, in which a derivative is equated to a function of state variables and exogenous variables. No restriction is placed, however, on the type of functions used; they may be linear or non-linear, with or without constraints. One very common type of constraint is imposed by the fact that most of the state variables (biomass, population, etc.) are in their very nature non-negative, so that the derivative must be non-negative where the value of a state variable is zero.

Though differential equations are the commonest way of representing changes in the model, functions involving discontinuities (such as may be imposed by threshold values of influencing variables) or representing discrete processes are fully acceptable. The general structure of the models is also fully compatible with the introduction of stochastic elements in one or more of the submodels, as well as in exogenous variables.

The Computer Implementation

The computer representation of these models is written in FORTRAN IV. The intention has been to avoid features of FORTRAN IV which might be peculiar to specific machines or installations so that the models developed might be widely usable.

The computer programs include a general calling program, which handles much of the input and output, and puts together the results provided by a number of process subroutines. Each of the latter represents a submodel, and they may be nested if the submodels are further divided. There are also subroutines for parts of the input/output operations. Output takes the form of tabulations of the values of all state variables at particular dates, together with graphs of the time course of particular state variables through the period of simulation. The programs are written as far as possible, in general terms, so that they can be applied with minimal modification to a wide variety of ecosystems. In particular, the number of classes in each cross-classification of the data, and their designations, are left to be decided at execution time, and facilities are also provided for specifying or modifying the parameters of the system then.

For computer solution, the differential equations expressing the rates of changes in the state variables are replaced by difference equations over a time unit fixed for the submodel, but not necessarily uniform, over all submodels. If the approximation by difference equations over this time unit leads to negative values of an essentially non-negative variable, the program reduces the time unit as required.

Besides the changes in state variables, the program also keeps a record of the total exchange of the ecosystem with its environment, in terms of each of the constituents (chemical elements, water, and/or energy) included among the state variables, and distinguishes between exchanges with the atmosphere, with the subsoil or by overland flow or exchange with laterally adjacent areas.

DESCRIPTIONS OF MODELS

Standardized descriptions of the programs and submodels are being prepared for distribution. Though the computer programs implementing the models will be included in these descriptions, their primary purpose is to describe the model itself in verbal and mathematical terms rather than the computer program -- which can speak for itself to those who are interested in the implementation as well as in the conceptualization.

After a brief introduction, the description will detail the assumptions incorporated into that particular model, and will then describe the way in which the various processes are treated. For each process, a verbal description will be followed by a mathematical representation of the differential (or other) equations incorporated in the model. For these mathematical representations (which can be skipped by readers who are not mathematically oriented), a standard symbolism is being developed. Since the number of distinct variables and parameters required with proliferation of further submodels may be vary large, it is not expected that it will always be possible to use consistently the same symbols for the same variables (or parameters) in all models, though this will be done as far as possible. Consistency is however, being sought in respect of the classes of symbols, and in the use of sub-scripts, as follows:

1) State variables are designated by X, subscripted to indicate the particular state variable in question. It is intended to reserve $X_1 \ldots\ldots X_9$ for state variables within the plant sub-system, $X_{11} \ldots\ldots\ldots X_{19}$, for those within the animal subsystem, and $X_{21} \ldots\ldots$ for those concerned with micro-organisms, or non-living components of the system. It will be convenient to consider the exterior as specified by a series of dummy state variables, whose absolute values may be meaningless, but changes in which represent the exchanges of the ecosystem with its environment. These dummy state variables will be represented by $X_0 \ldots\ldots X$.
2) Rates of change in state variables are represented by a superposed dot, as:

$$\dot{X}_2 = \frac{dX_2}{dt}$$

3) Parameters of equations in the system -- values not changed by the system, though sometimes varying in the step-wise fashion -- are indicated by a P

4) Exogenous variables are signalled by a V, for instance:

$$V_{12}$$

5) Temporary variables: variables required in the course of calculation, or for purposes of explaining an algorithm -- are designated by a subscripted Z, as

$$Z_2$$

6) Output variables -- those calculated from state variables for output purposes only, and playing no part in the dynamics of the system -- are signalled by Y, as:

$$Y_3$$

7) Sub-division of the rate of change of a state variable -- usually, a particular flux -- is indicated by an italic capital used as prior subscript. Thus:

$${}_{H}X_2$$

may represent that part of the change in X_2 which is attributable to herbivory, say -- that is, the direct flux from plant biomass to animal biomass.

8) Classes of variables or parameters are indicated by lower-case italic letters used as posterior subscripts. Thus, for a state variable classified in two ways (say, by animal species and cohort)

$$X_{11_{ac}}$$

would represent its value in the a'th species and the c'th cohort. The same subscripts may also be applied to parameters.

9) The following posterior subscripts have been standardized:

- a animal species or group
- c animal cohort
- d type of detritus
- f chemical fractions (constituents)
- g plant organ or portion
- m type of heterotrophic microorganisms
- p plant species or group
- r route of exchange at ecosystem boundaries

10) For certain of these subscripts, different values have meanings which have been standardized in the descriptions of earlier versions of the submodels, as follows:

<u>Chemical fractions of constituents</u>

$f = 1$	Carbon
$f = 2$	Nitrogen
$f = 3$	Phosphorus
$f = 4$	Other ash elements
$f = 5$	Chemical energy (in Kcal. m^{-2})

<u>Detritus types</u>

$d = 1$	Fine organic particles
$d = 2$	Coarse organic particles
$d = 3$	Inorganic material

<u>Types of heterotrophic microorganisms</u>

$m = 1$	Planktonic
$m = 2$	Benthic
$m = 3$	Attached

<u>Route of Exchange</u>

$r = 1$	To or from atmosphere
$r = 2$	By lateral flow
$r = 3$	To or from sub-soil

11) Where it is useful to define a subset of subscript values, this subset is symbolized by an italic capital. Thus, of the set of chemical fractions or constituents indicated by the subscript f the subset containing the carbon fractions only is designated by

$$C = (3,4,5)$$

The subset of elements other than carbon is designated by

$$M = (1,2)$$

Operations limited to subset of values are indicated by the symbol ε thus,

$$Y_p = P_{1_p} X_{1_{pgf}} \quad , \qquad p \varepsilon A$$

indicates that this function applies only to cases where the subscript p is included in the subset A; and

$$\sum_{p \varepsilon A}$$

12) Some readers may not be familiar with the pi-product notation, parallel with the sigma notation for summation; thus,

$$\prod_{p=1}^{n} X_{1_{pgf}} \equiv \text{p}\, X_{1_{pgf}} \equiv X_{1_{1gf}} \cdot X_{1_{2gf}} \cdot X_{1_{3gf}} \cdots\cdots\cdots\cdots X_{1_{ngf}}$$

$$\text{and}\ {}_{p\,\varepsilon\,A}\ X_{1_{pgf}} \equiv X_{1_{2gf}} \cdot X_{1_{4gf}} \cdot X_{1_{7gf}}$$

where the subset A is defined as:

$$A = (2,4,7)$$

13) Other conventions used consistently include the following: exp is an abbreviation for "exponential"; i.e.

$$\exp\,(a) = e^{a}$$

Ln is used for the natural or Napierian logarithm:

$$\text{In}(a) = \log_e a$$

Max is the abbreviation for "maximum" and min is an abbreviation for "minimum", thus:

$$\min(X_4, X_5)$$

indicates the smaller of the values X_4 and X_5 , while $\max(X_4, X_5)$ indicates the larger of the two values. A subscript placed below max or min:

$$\max(X_{12_{sf}}/Z_{9_{sf}})$$

indicates that the expression which follows it should be evaluated for all values of the subscript, and the largest (or smallest) of the resulting quantities taken.

The description of the processes in the model is followed by a list of the symbols used, with definitions and FORTRAN equivalents. There is then some discussion of constraints imposed on the model by the existing computer program, particularly by array dimensions, and changes which might be required to enable it to meet other needs. A program listing follows along with examples of data used, and of output. The time needed for running the program on a standard installation is also indicated, together with its storage requirement.

CALLING PROGRAM AND INPUT/OUTPUT SUBROUTINES

Joseph Wlosinski and David W. Goodall

2.1.3.1.2.-9

CALLING PROGRAM

INTRODUCTION

The computer program to be described in this report provides a common framework for the aquatic models to be developed. It does not itself model the dynamics of the system - a task performed by subroutines, which may be varied independently of the main program and of one another. The main program described below organizes most of the input operations, including the calculations of quantities required only as collective input to the subroutines, performs the incrementation of the state variables in accordance with calculations performed by the subroutines.

The program is designed to cover a wide range of aquatic ecosystems in which the state variables consist of the weight of various chemical constituents and energy contained in plant material, subdivided by species or species group, and by organ group (though this last facility is not used in the present implementation); animals, subdivided by species or species group and by stage of development; different types of detritus, suspended or in bottom sediments; and different types of heterotrophic microorganisms. The state variables also include the population of each animal group, and the composition of the water mass. Additional state variables may be introduced by the subroutines. The main program then treats them in the same way as the other state variables, and provides facilities for printing them out if wished. Exogeneous variables are acquired through the subroutine TERN, while output is organized by the subroutines REPORT and GRAF.

The parameters of the system (i.e. the constants incorporated in the equations expressing rates of change in the state variables) do not figure in this program, but are introduced into the programs implementing the process submodels, which are called as subroutines in the course of the main program discussed below.

This calling program, as stated above, is designed for use with a wide variety of process subroutines which may be developed in the future. The process subroutines in use at present, however, do not make use of certain of the options provided. For instance, they do not distinguish between different organ types in the primary producers, nor do they distinguish among the various chemical fractions into which the carbon content of the organisms may be divided.

INPUT ORGANIZATION

The successive cards required for input, many of which are optional and determined by the special requirements of the model in question, are detailed below. Constraints are placed on these input data by the array sizes of the program as compiled. These contraints are described in a later section of this report.

I. COMMENTS AND TABLE HEADING

Any comments which it is wished to associate with the output may be printed out before the rest of the output by inserting cards bearing the comment information at the beginning of the input deck. These cards should finish with a blank, or be replaced by a blank if no comments are needed. The blank ending the comments is followed by a single card providing a heading for tabular output.

II. INSTRUCTIONS CARDS

a. Dimensions, Specifications, Input/Output Instructions and Switches

The next three (+) cards contain (in (16I5) format) the following information in successive fields of five columns, right-justified:

Card A

1. The number of plant species or species groups.
2. The number of animal species or species groups.
3. The number of plant organs or organ groups distinguished.
4. The number of carbon fractions separated.
5. The number of chemical "elements" (including energy, but excluding carbon if the figure in IIa (4) was positive.)
6. The number of types of detritus distinguished; the same list of detritus types is assumed to apply both to suspended detritus and that constituting the bottom sediments.
7. The number of different names to be read in for animal cohorts or stages of development.
8. The number of different types of heterotrophic microorganisms.
9. The time step for simulation (in thousandths of a day); if this field is non-positive, the time step is taken as one day.
10. The number of entries in the "instructions" array (see IIb below) to be passed to the subroutines.
11. The number of entries in the "Repetitions" array, determining the time units for the subroutines (see IIc below); if this value is zero, all subroutines are assumed to use the same time unit as the main program - that specified in (9) above.

Card B

12. The starting date of the run, counting from the beginning of the year.
13. The number of the day on which the run is to finish, counting from the beginning of the year in which the run starts.
14. The number of tabulated reports required after the initial report.
15. The number of line graphs required.
16. The number of block graphs required.
17. The year in which the run is to begin.

If the value at (14) is positive, one or more further cards in format (16I5) are read in giving the dates (calculated from the beginning of the year in which the simulation starts) at which tabulated reports will be required. The number of such entries will be equal to the figure in (14). If (14) is blank, only initial and final reports will be provided (but see 22 below).

Card C

18. A switch for debugging purposes; if this is positive, extra information is printed out by many of the subroutines in the course of their operation, from the day of simulation it specifies.
19. A switch to complete the debugging operation begun under the previous instruction. If this value is less than that in the previous field, the debugging operation will continue to the end of the run.
20. A switch for timing purposes; if this switch is zero, no timing information will be included in the output; if it is set at "1", timing information will be given for each report and graph produced; if it is set at "2", the C.P.U. time for each time unit simulated will be reported.
21. A switch which must be positive if sensitivity tests are to be performed.
22. A switch for tabular reports:
    - 0: Initial and final reports, together with reports on any intermediate dates specified.
    - 1: Only the initial report is required.
    - 2: All tabular reports are to be omitted.
    - 3: The initial report is to be omitted.
23. A switch to provide (when positive) for the printing, with the tabulated reports, of any additional state variables initiated by the process subroutines and stored in the array DUMMY.
24. This and the next field give facilities for a portion of the parameter list to be printed before simulation starts. If the value in this field is positive, it causes values in the block / PARAM /, from this address onwards, to be printed out as soon as they have been read in by the process subroutines.
25. This field gives the last address for values in the COMMON block / PARAM / to be printed out under control of the switch in IIa (24) above.
26. A switch to provide (when positive) for the state variables to be read (in binary form) from a mass storage file designated as Unit 9, instead of from cards.
27. A switch to provide for the state variables to be dumped at specified times, in binary form, into mass storage files designated as Unit 10, Unit 11, etc. The number of such dumps is punched in this field.

28. If the previous field (IIa (27)) is occupied, the dates (from Jan. 1 (etc) in the first year of simulation) on which dumps are to be made are specified in these fields. The dates must be in order, and the number of fields occupied is equal to the number in IIa (2).

b. Instructions to Subroutines

If IIa (10) is positive, a number of integers equal to the value in this field is read in, in (16I5) format. These entries may be used for communicating with subroutines at execution time, and conveying instructions modifying their mode of operation.

c. Repetitions of Subroutines

If IIa (11) above was occupied by a positive value, a series of cards equal in number to the value in this field is read in. Each of these cards has in the first field of five columns a number, right-justified, representing one of the subroutines ("1" for VEGET, "2" for ANIMAL, and "3" for MEDIUM; other designations will be alloted later as required); the second field of five columns (6-10) contains, similarly right-justified, the number of times this subroutine is to be repeated within each of the time units simulated - or, in the case of a nested subroutine, within each operation of the subroutine which calls it. In other words, this provides a facility for varying the time units within subroutines, but limited to integral submultiples of the time unit used within the main program.

III. STAGES OF DEVELOPMENT

a. If the value in IIa (7) is greater than one:

1. A card is read in with the number of distinct stages of development for each species of animal, in (16I5) format. The number of entries should equal the number of animal species or groups in IIa (2).
2. This card is followed by one card for each species defined in IIIa (1) as having more than one stage of development, and less than the maximum number specified in IIa (7). Each of these cards contains, in (16I5) format, the numerical designations of these stages of development, corresponding with the names to be used for them in output (see IVe below).

IV. NAMES

a. If plants occur in the system (if the value at IIa (1) is positive), the names of plant species or groups are read in, two to a card, with up to 28 characters for each (i.e., the fields used are columns 1-28, and columns 29-56). The number of these names should correspond with the value in IIa (1).

b. If there are animals present (if the value at IIa (2) is positive), the names of animal species are read in, in the same way as those for plant species. They should correspond in number with IIa (2).

c. If plant organs are to be distinguished (i.e., if the value at IIa (3) was positive) the names of these organs or organ groups are read in, three to a card, with up to 24 characters for each (i.e., the fields used are columns 1-24, 25-48 and 49-72).

d. The names of chemical constitutents are read in, up to 12 characters for each, and in (20A4) format. Carbon fractions should follow the other constituents. In the present implementation, where carbon fractions are not distinguished, carbon has the first place and energy the last place in the list. The total number of constituents is the sum of the numbers at IIa (4) and IIa (5).

e. If the value at IIa (7) is greater than one, the names of the stages of development for animals are read in, five to a card, each with up to 16 characters (i.e., the fields used consist of columns 1-16, 17-32, 33-48, 49-64, and 65-80). The number of entries is equal to the value at IIa (7).

f. If the value at IIa (6) is positive, the names of different types of detritus are read in, following the same format as for stages of development (IVe above). The number of entries is equal to the value in IIa (6).

g. If the value at IIa (8) is positive, the names of different groups of heterotrophic microorganisms are read in, following the same format as for stages of development (IVe above). The number of entries is equal to the value at IIa (8).

V. INITIAL VALUES OF STATE VARIABLES

The cards in this section give values for the state variables at the starting point of the simulation. Most of them contain the quantities of the various chemical constituents per unit area ($g.m^{-2}$) in the same sequence as in IVd. All are in (8F10.4) format.

a. For each plant species group and each of the organ types, a card is read in with the initial values of chemical constituents per unit area of water surface (i.e., $g.m^{-2}$). The first species is taken first, and cards equal in number to the number of organs (IIa (3) above) - assumed one, if this entry was blank - are read; these are followed by a similar set of the second species, and so forth. The total number of such cards should accordingly be the value at IIa (1) multiplied by that at IIa (3) if positive.

b. For each animal species a series of cards is read in giving the population and biomass. The first cards for each animal species contains the population figures for successive stages of development (the number of such figures being equal to the corresponding entry in IIIa (1)). A card or cards follow in which the minimum weight of carbon (as a measure of biomass) for a single individual among those at this stage of development when the simulation begins. Finally, there is a sequence of cards, each containing the quantity of chemical constituents for one of the successive stages of development.

c. If the figure at IIa (8) is positive, a series of cards is read in for the quantity of chemical constituents in the different types of heterotrophic microorganisms, one for each type, equal in number to the value at IIa (8).

d. One card is read for each of the detritus types, giving the quantity of the various chemical constituents in suspended material of this type, per unit area of water surface. The total number of such cards is equal to the figure in IIa (6).

e. The quantity of chemical constituents in the different types of detritus in the bottom sediments is read in, in a series of cards parallel with those for suspended detritus in Vd.

f. A card is read in giving the quantities of dissolved chemical elements in the water mass, per unit area of surface. The fields occupied are one more than the figure in IIa5, the last being for bicarbonate-carbon.

VI. SPECIFICATIONS FOR GRAPHICAL OUTPUT

If the value at IIa (15) is positive, a series of cards are read in specifying the line graphs required. For each of the graphs in succession, the following cards are needed:

1. A card specifying which variables are to be graphed. These are expressed as addresses in the state variables array (COMMON block / STAT /); the addresses in the sums array (COMMON block / TOTALS /) increased by 10,000 the addresses in the external exchange array (COMMON block / ACC /) increased by 20,000; or the addresses in the array of additional accessible variables (COMMON block / OTHER /) increased by 30,000. These addresses are punched in (16I5) format, and may not exceed eight in number for each line graph.
2. A title card for the graph; all 80 columns may be used.
3. A title for the Y-axis of the graph (the X-axis always being in days). This title may occupy columns 1-40 of the card. If the word "ZERO" is punched in columns 41-44, the Y-axis of the graph will include zero; otherwise, it will extend from the minimum to the maximum value of the variables graphed.
4. If VIa (1) designates more than one variable, these are followed by a series of cards equal in number to the entries in VIa (1). Each card gives a brief explanation of one of the variables included in the graph, in the same order as their addresses are listed in VIa (1). Twenty characters are allowed for each, which should be in columns 1-20 of the card.
5. If the value at IIa (16) is positive, a card is read in specifying the variables for which block graphs are required; these addresses are coded according to the same rules as in VIa (1). This card is followed by two cards for each of the block graphs, the first being a title, the second a title for the Y-axis, limited to the first 40 columns of the card.

VII. INPUT REQUIRED BY SUBROUTINES

a. Each of the process subroutines may require parameters, and other specifications to be read in. This takes place after all the preceding input requirements have been met. Where such information is needed, reading is performed by a NAMELIST statement. The first card for each reading operation begins with

b$NAMEb

where NAME represents the name of that NAMELIST in the subroutine in question, and b represents a blank column. This and subsequent cards then contain entries in the forms:

A = *a* , B (3) = *b* , C = *c* , *d* , *e* , n * *f* ,

where A is the name of a variable and B and C are names of arrays included in the NAMELIST, *a*, *b*, *c*, *d*, *e*, *f*, are constants of the appropriate type, and n is an integer. Each NAMELIST input is concluded with

b$END

For this purpose, the process subroutines are called in the order MEDIUM, ANIMAL and VEGET.

b. Specifications for sensitivity tests are then read in. These are discussed in the report on the sensitivity subroutines.

c. Finally, information on the exogeneous variables is read in. This input is discussed in connection with the EXTERN subroutine below.

Computational Operations

The central part of the program is responsible for incrementation of the state variables. When calculation of all the increments over a single time unit (of which submultiples may be used for some of the subroutines) has been completed, they are tested to ensure that none of them would cause state variables to become negative, where this constraint is appropriate (which is true of most state variables in ecological systems). If some of the negative increments are "too large" in this sense, all increments are scaled down in such proportions as the most limiting constraint requires, the increments are applied to all state variables, and the subroutines are called again for recalculation of increments. These increments are then multiplied by the complement of the proportion already applied to the state variables, and the test of their magnitude is repeated. The process continues until a set of increments can be applied *in toto*. Briefly, this is equivalent to dividing the time unit over which the difference equations approximate the underlying differentials into arbitrary portions such that the constraints can be met. This process, central to the program, may be represented as follows: Let X_{ij} be the value of the i'th state variable at the beginning of the j'th iteration, and ΔX_{ij} the increment for one time unit as calculated by the subroutines for that iteration. Then:

$$X_{i,j+1} = X_{i,j} + t_j X_{i,j}$$

where:

$$t_j = -(\min(-1, \frac{\Delta X_{ij}}{X_{ij}}))^{-1}$$

Iteration is completed when: $\sum_j \; t_j = 1$

Exchanges between the ecosystem and its surroundings are accumulated by the calling program from data provided, time unit by time unit, by the subroutines. These quantities, together with any state variables not constrained to take non-negative values, are incremented in proportion to the increments mentioned in the previous two paragraphs, so that the whole program is operating in time units shorter than that prescribed whenever this proves necessary.

Sums of state variables for all classes and combinations of classes are required for output, and may also be needed by the subroutines. These summations are accordingly performed by the main program initially, and again after each time unit of the simulation has been performed. Exchanges with the surroundings of the ecosystem - with the air, with the subsoil, and by surface flow - are also accumulated after each time unit.

Array Dimensions

The use of the program is limited by the dimensions allotted to the arrays, and these limitations need discussion so that the user may be in a position to modify them as his particular requirements indicate. Below is a list of arrays included in the calling program, in which the dimensions which may appropriately be varied are indicated by letters. Dimensions of other arrays, and dimensions indicated by numbers, are subject to other constraints, and changes in them would call for other changes in the program.

ABACT (m)	AVEG (j,n)	CMIN (e,h)	CVEGO (j,h)	EXPLAN (5,g)	PLDEP (j)
ABIOM (a)	AVEGO (j)	CMINH (h)	CVEGQQ (j,n,h)	FIG (v,70)	ORGNAM (n,6)
ABIOSP (b)	AVEGV (n)	CMINQQ (e.h)	CVEGV (n,h)	FIGS (g,70)	POP (a)
AGAIN (c,d)	BACNAM (m,4)	COHNAM (p,4)	CVEGVO (h)	FRANAM (h,3)	POPQQQ (a)
AGAINQ (c,d)	BIOMIN (a)	COMPIN (h)	DADUST (e,h)	H2O (c)	POPSP (b)
AGAINT (c)	BIOMIQ (a)	CONNIT (j,k)	DECINC (r)	H2OQQQ (c)	PRODRF (j)
ALINAM (e,6)	CBACT (m,h)	CONNI2 (j,k)	DETIN (n,h)	INSTRU (y)	RAINCO (h,d)
ALIT (e)	CBACTQ (m,h)	CONRAD (j)	DRIFTA (a,h)	LIGRAF (g)	RESPC (j)
AMAXI (f)	CBACTT (h)	CONTE1 (j)	DRIFTM (m,h)	LISCOH (a)	RUNDEB (e,h)
AMINI (f)	CBIOM (a,h)	CONTE2 (j)	DRIFTV (j,n,h)	LISTER (f)	RUNSOL (d)
AMORT (j)	CBIOMA (h)	CONTE3 (j)	DUMMQ (o)	LITRUN (e)	SIEVEG (j,h)
ANIM (b,h)	CBIOMQ (a,h)	CORG (e.h)	DUMMY (o)	MREP (z)	STATE (4)
AORG (e)	CHNG (x)	CORGH (h)	DUSCOM (h)	NCOH (b)	STNG (x)
AQUA (h)	CLIT (e,h)	CORGQQ (e,h)	EROD (c)	NCOHCU (b)	SUMS (w)
AQUAQQ (h)	CLITT (h)	CORGT (h)	ERODQQ (c)	NREPET (u)	TITLES (f,20)
ASPNAM (b,7)	CLITTQQ (e,h)	CVEG (j,n,h)	EXOG (i)	PHOTA (j)	TOT (h)
			EXPLA (5,v)	PHOTD (j)	

TOTAL (h)
UPCON (j,k)
UPCON2 (j,k)
VSPNAM (j,7)
YAXISS (f,10)

The dimensions indicated by letters define the maximum values possible for the following quantities:

		FORTRAN
a:	Total number of animal cohorts.	NSPCOH
b:	Number of animal species (or groups).	NSPECA
c:	Number of channels for gain or loss to or from the system.	NCHAN
d:	Number of separate components tracked (including energy, but excluding separate forms of any element where appropriate).*	NELEM
e:	Number of types of detritus.	NOLIT
f:	Number of separate graphs.	NOHISU
g:	Number of variables to be graphed.	
h:	Total number of constituents tracked (including energy where appropriate, and the separate forms of any element).*	NFRELM
i:	The dimension of the dummy array EXOG, required for initialization of the Common Block METEOR in the EXTERN subroutine, q.v.	
j:	Number of plant species (or groups).	NSPECV
k:	Number of chemical elements *sensu stricto* (excluding energy)	
m:	Number of types of heterotrophic microorganism.	MICROB
n:	Number of plant organ types.	NORGAN
o:	Extra addresses for state variables and increments available to process subroutines in arrays DUMMY and DUMMQ.	
p:	Number of different names for animal cohorts.	NCOHOR
r.	Total number of "words" in Common Blocks STAT and CHANGE; these dimensions should equal the value defined for LIMIT.	LIMIT
u:	Number of process subroutines.	NOTIME
v:	Number of curves on a single graph.	
w:	Total number of words in Common Block TOTALS; this dimension should equal the value defined by LIMTOT.	LIMTOT
x:	Total number of "words" in Common Blocks ACC and ACCINC; this dimension should equal the value defined for LIMACC.	LIMACC
y:	Number of instructions to be transferred to subroutines.	NOINST
z:	Number of tabulated reports to be provided.	NREP

* At present, no separate forms of any element (e.g., carbohydrate carbon and fat carbon, nitrate nitrogen and protein nitrogen) are included in the models, and consequently d and h may remain identical.

It should be noted that, if the dimensions of any of the arrays in the common blocks are changed, not only must these blocks be changed in any subroutines where they occur, but the arrays equivalenced with the common blocks must be changed to correspond, as also the variables specifying their limits, thus:

Common Block	Array	Limit
ACC ACCINC	STNG CHNG	LIMACC
TOTALS	SUMS	LIMTOT
STAT CHANGE	STATE DECING	LIMIT

Arrays occurring in the paired common blocks ACC and ACCINC, STAT and CHANGE must correspond in their order, and in all their dimensions, as tabulated below:

Arrays that must match in their dimensions	
AGAIN	AGAINQ
AQUA	AQUAQQ
BIOMIN	BIOMIQ
CBACT	CBACQ
CBIOM	CBIOMQ
CLIT	CLITQQ
CMIN	CMINQQ
CORG	CORGQQ
CVEG	CVEGQQ
DUMMY	DUMMQ
EROD	ERODQQ
H2O	H2OQQQ
POP	POPQQQ

SUBROUTINE EXTERN

This subroutine provides for the initial input of meteorological and other exogeneous data and for the supply of daily values of these variables to the process subroutines requiring them. The input required by this subroutine occurs after all other inputs. Unless otherwise stated, all cards are in (8F10.2) format. The cards required are as follows:

1. The mean daily quantity of water (in $g.m^{-2}$) flowing into the system.
2. The mean concentration (in grams per gram) of dissolved chemical constituents in the inflowing water.
3. The mean quantity of chemical constituents in suspended material in the inflowing water (in grams per gram of water) one card for each detritus type (cards equal in number to the value at IIa (6) in the description of the calling program.

4. Detritus falling into the system from above. For each month of the year, a series of cards equal in number to the value at IIa (6) above is read in, providing figures for the quantity of each chemical constituent (in g.m^{-2} day) contained in a particular detritus type.
5. In format (16I5), values for:
   a. Number of days for which precipitation is recorded.
   b. Number of days for which surface flow from adjoining land surfaces is recorded.
   c. Number of days on which water is withdrawn for irrigation or other purposes.
6. In format (16I5), the dates (from the beginning of the starting year) of precipitation events.
7. The quantities of precipitation (in mm. water) on the dates specified in (6).
8. The mean composition of precipitation (in g. per g), for the same elements and in the same order as for the ecosystem components in the main program. If the figure in 5b is positive, the next cards contain:
9. In format (16I5), the dates (from the beginning of the starting year) of surface flow events.
10. The quantities (in g.m^{-2}) of water imported as surface flow on the dates specified in (9).
11. The quantities (in g.m^{-2}) of eroded soil imported with surface flow for each date specified in (9).
12. For each date specified in (9), a pair of cards giving:
    a. The amounts (in m^{-2}) of chemical elements in inorganic form imported with the surface flow.
    b. The amounts (in g m^{-2}) of chemical constituents in detritus imported with the surface flow. If the figure in 5c is positive, the next cards contain:
13. In format (16I5), the dates (from the beginning of the starting year) on which water was withdrawn.
14. The quantities of water withdrawn on the dates specified in (13). The following cards are included on all occasions:
15. Two cards giving the monthly mean pan evaporation per day (in mm.).
16. Two cards giving the monthly mean photoperiod (in hrs.).
17. Two cards giving the monthly mean radiation intensity (cal. cm.$^{-2}$min^{-1}).
18. Two cards giving the monthly mean water temperature (in C.).

Operations

After input cards have been read in, control returns to the main program. The subroutine is called again at the entry point EXOGE2 whenever a new day is simulated If rainfall, surface flow, or water withdrawal occurs on that day, the appropriate values are transferred from storage to the METEOR common block. Similarly, every time the month changes the appropriate monthly means for the other meteorological variables are transferred.

ARRAY DIMENSIONS

Limitations imposed by dimensions for arrays used in the EXTERN subroutine, but not in the main program, are as follows:

DUST (12,g,f)
ERODED (b)
EXO (c)
MIRRIG (e)
MRAIN (a)
MRUNON (b)
RAIN (a)
RUNMIN (b,d)
RUNON (b)
RUNORG (b,g,f)
WATIRR (e)

Where:		FORTRAN
a:	Number of days with precipitation during the period simulated.	NRAIN
b:	Number of days with surface flow during the period simulated	NRUNON
c:	Total number of words in Common Block METEOR; this dimension should equal the value defined for LIMEXO.	LIMEXO
d:	Number of separate components tracked (including energy where appropriate, but excluding separate forms of any element).	NELEM
e:	Number of days on which water is withdrawn during the period simulated.	NIRRIG
f:	Total number of constituents tracked (including energy where approriate, and the separate forms of any element).	NFRELM
g:	Number of categories of detritus.	NOLIT

Other dimensions are constrained by other features of the program, and may not be varied so readily.

N.B. If any dimensions within the Common Block METEOR are changed, that of EXO with which it is equivalenced must also be changed to correspond; a value equal to the dimension of EXO must be given to LIMEXO, and the value of LIMEX1 should be the address of ERO in the Common Block METEOR.

SUBROUTINE REPORT

At times when tabulated output has been requested, the values of all state variables are printed together with their sums.

The tabulated output also includes, on all occasions except the initial report, a table of the quantities of chemical elements, of water, and of inert soil particles lost by the ecosystem to its surroundings, or gained as the case may be. Exchanges with the atmosphere, with the subsoil (below the lowest horizon for which state variables are included), and laterally along the soil surface, are distinguished. These figures are accumulated over the period of simulation. Precipitation figures are also accumulated and included in the reports. The depth of snow lying on the soil surface is reports in mm., and its weight in $g.ha^{-1}$.

Array Dimensions

Limitations imposed by dimensions for arrays used in the REPORT subroutine, though not in the calling program are as follows:

SOURCE (a,b)   AGAINT (a)

a: Number of separate components (chemical elements, energy) exchanged with the surroundings.

b: Number of channels for gain or loss to the system.

SUBROUTINE GRAF

Graphical output is available on the line printer, in the form of either line or block graphs, in each case occupying a single printer page. Line graphs use continuous strings of the symbols A,B, ... H, to represent the time course, through the simulation, of up to eight different variables. All are plotted on the same scale, which is adjusted so that the extreme values attained can just fit into the page. In the block graphs, the time course of a single variable only is plotted. Each type of graph is provided with appropriate titling.

The X-axis of the graph is always the time in days. The Y-axis is scaled so that all values graphed are within the range $-10 < y < +10$, and an integral power of 10 is specified in the title as the scaling factor. If the values extend outside the limits $\pm 10^{10}$, an error message results. If all values of Y are identical, a small range of Y values around this point is graphed.

The graphs are printed after the final tabulated report. Graphing facilities are provided for all state variables; for all their totals; and for all accumulated exchanges between the ecosystem and its surroundings.

TIME AND SPACE REQUIREMENTS

Facilities are provided for monitoring the C.P.U. time required for different parts of the simulation. For this purpose use is made of the system subroutine EXTIME, which would need to be replaced if the programs were implemented on any other system. As compiled on the UNIVAC 1108, the programs listed require 17 seconds for compilation; the coding occupies 11,000 36-bit words of core storage and the storage of variables (with the array sizes specified) required a further 41,000 words.

Operation of the program required about 9 seconds and 54,000 words storage for the example given.

```
C    AQUATIC    MAIN

C ABIOMA         CONTENT OF CARBON (ALL FRACTIONS) TOTALED OVER ALL
C                ANIMAL COHORTS
C ABIOSP(K)      CONTENT OF CARBON (ALL FRACTIONS) IN ALL COHORTS OF
C                THE K'TH ANIMAL SPECIES
C AGAIN(P,N)     NET CHANGE IN THE N'TH CONSTITUENT IN THE SYSTEM AS A
C                WHOLE THROUGH THE P'TH CHANNEL (ATMOSPHERE, SURFACE
C                FLOW, SUBSOIL FLOW)
C AGAINQ(P,N)    CHANGE PER TIME UNIT IN AGAIN(P,N)
C AGAINT(N)      NET CHANGE IN THE N'TH CONSTITUENT IN THE SYSTEM AS A
C                WHOLE
C ALTNAM(M,A)    NAME OF THE M'TH DETRITUS FRACTION, UP TO 16
C                CHARACTERS
C ALTT(M)        CONTENT OF CARBON (ALL FRACTIONS) IN THE M'TH
C                CATEGORY OF SUSPENDED DETRITUS
C ALTT'          CONTENT OF CARBON (ALL FRACTIONS) IN THE SUSPENDED
C                DETRITUS
C AMAXT(I)       MAXIMUM VALUE OF Y AXIS FOR I'TH GRAPH.
C AMICRO         VALUE BELOW WHICH STATE VARIALBES WILL BE TREATED AS
C                ZERO
C AMINT(I)       MINIMUM VLAUE OF Y AXIS FOR I'TH GRAPH.
C ANIM(K,N)      CONTENT OF THE N'TH CHEMICAL CONSTITUENT IN ALL
C                COHORTS OF THE K'TH ANIMAL SPECIES GROUP
C AORG(M)        CONTENT OF CARBON (ALL FRACTIONS) IN THE M'TH
C                CATEGORY OF BOTTOM DETRITUS
C AORGT          CONTENT OF CARBON (ALL FRACTIONS) IN THE BOTTOM
C                DETRITUS
C AQUA(N)        CONTENT OF THE N'TH CHEMICAL CONSTITUENT IN THE WATER
C AQUAQQ(N)      CHANGE PER TIME UNIT IN AQUA (N)
C ASPNAM(K,A)    NAME OF THE K'TH ANIMAL SPECIES (UP TO 28 CHARACTERS)
C AVEG(I,J)      CONTENT OF CARBON (ALL FRACTIONS) IN THE J'TH ORGAN
C                OF THE I TH PLANT SPECIES GROUP
C AVEGO(I)       CONTENT OF CARBON (ALL FRACTIONS) IN THE I'TH PLANT
C                SPECIES GROUP, TOTALED OVER ALL ORGANS
C AVEGV(J)       CONTENT OF CARBON (ALL FRACTIONS) IN THE J'TH ORGAN,
C                TOTALED OVER ALL PLANT SPECIES GROUPS
C AVEGVO         CONTENT OF CARBON (ALL FRACTIONS) TOTALED OVER ALL
C                PLANT SPECIES GROUPS AND ORGANS
C BLANK          STORED BLANK FOR HEADINGS
C CARBOT         HEADING FOR CARBON COLUMNS
C CARBON         HEADING FOR CARBON COLUMNS
C CBIOM(L,N)     CONTENT OF THE N'TH CHEMICAL CONSTITUENT IN THE L'TH
C                ANIMAL COHORT
C CBIOMA(N)      CONTENT OF THE N'TH CHEMICAL CONSTITUENT TOTALED
C                OVER ALL ANIMAL COHORTS
C CBIOMQ(L,N)    CHANGE DURING THE TIME UNIT IN CBIOM(L,N)
C FRAC(A)        HEADER FOR TABULATION OF CARBON FRACTIONS
C CHNG(A)        CHANGE IN A'TH ACCUMULATOR (IN COMMON BLOCK ACCINC)
C                (A.LE.LIMACC)
C CLIT(M,N)      CONTENT OF THE N'TH CHEMICAL CONSTITUENT IN THE M'TH
C                CATEGORY OF FLOATING DETRITUS
C CLITQQ(M,N)    CHANGE DURING THE TIME UNIT IN CLIT(M,N)
C CLITT(N)       CONTENT OF THE N'TH CHEMICAL CONSTITUENT IN THE
C                FLOATING DETRITUS
C COHNAM(L,A)    NAME OF THE COHORTS DESIGNATED AS 'L' (UP TO 16
```

```
C                   CHARACTERS)
C COMPTN(N)          CONTENT (IN G.PER G.) OF THE N'TH CONSTITUENT IN
C                   SOLUTION IN THE INFLOWING WATER
C CORG(M,N)         CONTENT OF THE N'TH CHEMICAL CONSTITUENT IN THE M'TH
C                   CATEGORY OF BOTTOM DETRITUS
C CORGT(N)          TOTAL CONTENT OF THE N'TH CHEMICAL CONSTITUENT IN
C                   THE BOTTOM DETRITUS
C CORGQQ(M,N)       CHANGE DURING THE TIME UNIT IN CORG(M,N)
C CVEG(I,J,N)       CONTENT OF THE N'TH CONSTITUENT IN THE J'TH ORGAN OF
C                   THE I'TH PLANT SPECIES GROUP
C CVEGO(I,N)        CONTENT OF THE N'TH CHEMICAL CONSTITUENT IN THE I'TH
C                   PLANT SPECIES GROUP SUMMED OVER ALL ORGANS
C CVEGQQ(I,J,N)     CHANGE DURING THE TIME UNIT IN CVEG(I,J,N)
C CVEGV(J,N)        CONTENT OF THE N'TH CHEMICAL CONSTITUENT IN THE J'TH
C                   ORGAN, SUMMED OVER ALL PLANT SPECIES GROUPS
C CVEGVO(N)         CONTENT OF THE N'TH CHEMICAL CONSTITUENT, TOTALED
C                   ALL PLANT SPECIES GROUPS AND ORGANS
C DECINC(A)         CHANGE IN A'TH STATE VARIABLE (A.LE.LIMIT)
C EROD(P)           ACCUMULATED NET GAIN OR LOSS OF INERT PARTICLES
C                   THROUGH THE P'TH CHANNEL (SEE 'AGAIN' ABOVE)
C ERODQQ (P)        CHANGE PER TIME UNIT IN EROD (P)
C EXPLA(B,A)        IN A GRAPH, EXPLANATION OF THE A'TH VARIABLE - UP TO
C                   20 CHARACTERS
C EXPLAN(B,A)       EXPLANATION (UP TO 20 CHARACTERS) OF A'TH LINE IN
C                   MULTIPLE-LINE GRAPHS, TAKEN TOGETHER
C FIG(A,B)          IN A GRAPH, VALUE OF THE A'TH VARIABLE FOR THE B'TH
C                   COLUMN
C FIGS(A,B)         VALUES OF A'TH VARIABLE FOR 71 SUCCESSIVE COLUMNS OF
C                   ALL GRAPHS, TAKEN TOGETHER
C FLOWIN            RATE OF INFLOW OF WATER (C.PER SQ.M.PER DAY)
C FRANAM(N,A)       NAME OF THE N'TH CONSTITUENT, UP TO 16 CHARACTERS
C H2O(P)            NET GAIN OR LOSS OF WATER THROUGH THE P'TH CHANNEL
C                   (SEE 'AGAIN' ABOVE)
C H2OQQQ(P)         CHANGE PER TIME UNIT IN H2O(P)
C HIGH              ARBITRARILY HIGH VALUE TO INITIALIZE MINIMIZATIONS
C IDAY              THE NUMBER OF THE CURRENT TIME UNIT
C IDUMP(II)         THE VALUE OF IDAY AT WHICH THE NEXT DUMP OF STATE
C                   VARIABLES IS TO BE MADE.
C INSTRU(A)         SWITCH FOR THE A'TH INSTRUCTION TO BE TRANSFERRED TO
C                   SUBROUTINES.
C IPARAM            THE ADDRESS IN THE COMMON BLOCK /PARAM/ FROM WHICH A
C                   PRINT-OUT IS TO START
C ISENCE            SWITCH, POSITIVE FOR SENSITIVITY TESTS
C IYR               STARTING YEAR
C IYRDAY            CURRENT TIME UNIT, COUNTING FROM JANUARY 1.
C JDAY              INITIAL DAY OF SIMULATION
C JPARAM            THE ADDRESS IN THE COMMON BLOCK /PARAM/ AT WHICH A
C                   PRINTOUT IS TO END.
C JSTATE            SWITCH TO PERMIT ARRAY 'DUMMY' TO BE PRINTED.
C KDUMP             THE FINAL ADDRESS IN IDUMP.
C LDEBUG            VALUE OF IDAY AT WHICH EXTRA OUTPUT IS TO CONCLUDE
C LIGRAF(A)         DESIGNATION NUMBER OF THE A'TH VARIABLE TO BE GRAPHED
C                   (ADDRESS IN ARRAY 'STATE , OR IN ARRAY 'SUMS' +
C                   10000, OR IN ARRAY 'STNG' + 20000)
C LIMACC            SIZE OF ARRAYS STNG,CHNG
C LIMTT             SIZE OF ARRAYS STATE, DECINC
C LIMTOT            SIZE OF ARRAY SUMS
```

```
C LTSCOH(L)        THE NUMERICAL DESIGNATION OF THE L'TH ANIMAL COHORT
C LSITEP(A)        NUMBER OF CURVES TO BE INCLUDED IN THE A'TH GRAPH
C MDEBUG           VALUE OF IDAY FROM WHICH EXTRA OUTPUT IS TO BEGIN
C MRAIN(I)         TIME UNITS (FROM BEGINNING OF FIRST YEAR) ON WHICH
C                  RAIN FALLS
C MREP (A)         DATE OF A'TH REPORT
C NCHAN            NUMBER OF CHANNELS FOR EXCHANGE WITH SURROUNDINGS
C NCHECK           SWITCH TO INDICATE FIRST USE OF SUBROUTINES
C NCHOH(D)         NUMBER OF DEVELOPMENTAL CATEGORIES (COHORTS) OF THE
C                  K'TH ANIMAL SPECIES
C NCOHCU(K)        THE STARTING ADDRESS IN LTSCOH FOR THE K'TH ANIMAL
C                  SPECIES
C NDAY             FINAL DAY OF SIMULATION
C NELEM            NUMBER OF CONSTITUENT ELEMENTS OR GROUPS OF ELEMENTS
C NELEM3           3*NELEM
C NEWBFR           A SWITCH, A POSITIVE VALUE OF WHICH PERMITS THE STATE
C                  VARIABLES TO BE READ FROM LOGICAL UNIT 9 RATHER THAN
C                  FROM CARDS.
C NFRAC2           NFRACT+2
C NFRAC1           NELEM + 1 IF NFRACT = 0 OR NFRACT + NELEM + 1
C                  IF NFRACT IS GREATER THAN 1
C NFRACT           NUMBER OF CARBON FRACTIONS
C NFRELM           NELEM + NFRACT
C NOCOL            NUMBER OF COLUMNS IN A HISTOGRAM
C NOGRAF           NUMBER OF LINE GRAPHS REQUIRED
C NOHTST           NUMBER OF BLOCK GRAPHS REQUIRED
C NOINST           NUMBER OF INSTRUCTIONS TO BE TRANSFERRED TO
C                  SUBROUTINES
C NOLIT            NUMBER OF DETRITUS CATEGORIES
C NOREP            SWITCH FOR TABULATED REPORTS: 1 FOR OMISSION OF ALL
C                  BUT INITIAL REPORTS, 2 FOR OMISSION OF ALL, 3 FOR
C                  OMISSION OF THE INITIAL REPORT ONLY
C NORGAN           NUMBER OF ORGANS FOR EACH PLANT SPECIES
C NOSECS           A SWITCH WHICH PROVIDES FOR TIMING OF OUTPUT
C                  OPERATIONS AND INCREMENTATIONS OF STATE VARIABLES.
C                  (0 CAUSES NO TIMING; 1 CAUSES TIMING OF OUTPUT
C                  OPERATIONS; 2 CAUSES TIMING OF OUTPUT OPERATIONS AND
C                  INCREMENTATION OPERATIONS)
C NOSYM            NUMBER OF VARIABLES IN A SINGLE GRAPH
C NOTIME           NUMBER OF SUBROUTINES WITHIN DO-LOOPS
C NRAIN            NUMBER OF OCCASIONS (IN ARRAYS MRAIN, RAIN) ON WHICH
C                  RAIN FALLS
C NREP             NUMBER OF TABULATED REPORTS
C NREPET(A)        NUMBER OF REPETITIONS FOR THE A'TH SUBROUTINE
C NSPCOH           TOTAL NUMBER OF DEVELOPMENTAL CATEGORIES (COHORTS)
C                  OF ANIMAL SPECIES
C NSPECA           NUMBER OF ANIMAL SPECIES CATEGORIES
C NSPECV           NUMBER OF PLANT SPECIES CATEGORIES
C NUNIT            TIME UNIT FOR SIMULATION IN THOUSANDTHS OF A DAY
C NYRDAY           NUMBER OF TIME UNITS IN A YEAR.
C ORGNAM(J,A)      NAME OF THE J'TH PLANT ORGAN (UP TO 24 CHARACTERS)
C POP(L)           POPULATION OF THE L'TH ANIMAL COHORT (NUMBER PER
C                  SQ. M.)
C POPQQQ(L)        CHANGE DURING THE TIME UNIT IN POP(L)
C POPSP(K)         POPULATION OF ALL COHORTS OF THE K'TH ANIMAL SPECIES
C                  GROUP (NUMBER PER SQ.M.)
C PRECMM           ACCUMULATED PRECIPITATION IN MILLIMETERS
```

```
C RAIN (A)          AMOUNT OF RAIN (IN MM.) ON THE A'TH TIME UNIT ON
C                   WHICH RAIN FALLS
C SOURCE(J,P)       NAME OF THE P'TH CHANNEL (SEE 'AGAIN' ABOVE) FOR
C                   GAIN OR LOSS TO THE SYSTEM, UP TO 24 CHARACTERS
C STATE(A)          A'TH STATE VARIABLE (A.LE.LIMIT)
C STNG(A)           A'TH ACCUMULATOR  IN COMMON BLOCK ACC   A.LE.LIMACC
C SUMS(A)           A'TH TOTAL (A.LE.LIMTOT)
C TIME              TIME EXPIRED SINCE LAST INQUIRY
C TITLE (A)         TITLE OF A GRAPH - (UP TO 80 CHARACTERS)
C TITLES (A,B)      TITLE OF THE A'TH GRAPH, UP TO 80 CHARACTERS
C                   (A.LE.20)
C TOT(N)            CONTENT OF THE N'TH CHEMICAL CONSTITUENT IN ALL
C                   DETRITUS
C TOTAL(N)          TOTAL OF N'TH CONSTITUENT IN WHOLE ECOSYSTEM.
C TOTNAT            HEADING FOR TOTAL COLUMNS
C TOTNAM            HEADING FOR TOTAL COLUMNS
C VSPNAM (I,A)      NAME OF THE I'TH PLANT SPECIES (UP TO 28 CHARACTERS)
C XDOT(A)           IN A BLOCK GRAPH, VALUE FOR THE A'TH COLUMN
C XMAX              MAXIMUM VALUE FOR X AXIS IN A GRAPH
C XMIN              MINIMUM VALUE FOR X AXIS IN A GRAPH
C XTITLE(J)         TITLE FOR THE X AXIS OF A GRAPH, UP TO 40 CHARACTERS
C YAXISS(I,J)       TITLE OF THE Y AXIS IN THE I'TH GRAPH (UP TO 40
C                   CHARACTERS)
C YDOT              NOT USED
C YMAX              MAXIMUM VALUE FOR Y AXIS IN A GRAPH
C YMIN              MINIMUM VALUE FOR Y AXIS IN A GRAPH
C YTITLE(A)         TITLE FOR THE Y AXIS OF A GRAPH, - UP TO 40 CHARACTERS
      COMMON /OTHER/PRECMM,TOTAL(5),CFRAC(5),SOURCE(6,3)
C-----------------------------------------------------------------------
C      THE COMMON BLOCK /NAMES/ CONTAINS THE NAMES REQUIRED FOR
C      TABULATED OUTPUT
C-----------------------------------------------------------------------
      COMMON /NAMES/COHNAM(21,4),BACNAM(3,4),VSPNAM(20,7),ASPNAM(20,7),
     1 ORGNAM(6,6),FRANAM(10,4),ALINAM(5,4)
      DIMENSION NEGATE(20),PLACE(20),DECJAN(12),LIGRAF(20)
      DIMENSION AMAXI(20), AMINI(20), FIGS(20,71), EXPLAN(5,20)         0060
      DIMENSION STATE (1586), DECINC(1586),SUMS(615)
      DIMENSION STNG(21),CHNG(21),MREP(20),MONDAY(12),       ORIGIN(30)
      DIMENSION TITLES(20,20),YAXISS(20,10),LISTER(10),IDUMP(5)
      DIMENSION MGRA(20)
C-----------------------------------------------------------------------
C      COMMON BLOCK /ACC/ CONTAINS ACCUMULATED CHANGES, WHICH MAY BE
C      NEGATIVE.  COMMON BLOCK /ACCINC/ CONTAINS THE INCREMENTS TO THE
C      ARRAYS IN /ACC/ FOR A SINGLE TIME UNIT.
C-----------------------------------------------------------------------
      COMMON /ACC/ AGAIN(3,5),FROD(3),H20(3)                            0180
      COMMON /ACCINC/ AGAINQ(3,5),FRODQQ(3), H20QQQ(3)                  0200
C-----------------------------------------------------------------------
C      COMMON BLOCK /SPEC/ CONTAINS SPECIFICATIONS AND OTHER INFORMATION
C      COMMON TO THE WHOLE SET OF PROGRAMS, BUT EXCLUDING STATE AND
C      EXOGENOUS VARIABLES.
C-----------------------------------------------------------------------
      COMMON/SPEC/TCOVER,NCHAN,INSTRU(20),    WATER,NSPECV,NSPECA,NORGAN,  0220
     1 NFRACT,NELEM,NOLIT ,NCHECK,IDAY, ATOT, ATOTO,IYRDAY,NREPET(20)      0240
     2,NCOH(20),LISCOH(98),NCOHCU(20),NCOHOR,NFRELM,NFRAC1,NSPCOH,NDEBUG   0260
     3,FLOUT,MICROB  ,BIOMIN(98)                                           0280
C-----------------------------------------------------------------------
```

```
C      COMMON BLOCK /TOTALS/ CONTAINS SUMS OF THE STATE VARIABLES,
C      TOGETHER WITH CERTAIN OTHER VARIABLES REQUIRING INITIALIZATION
C      BUT NOT INCREMENTATION AT EACH TIME UNIT.
C-----------------------------------------------------------------------
       COMMON/TOTALS/CVEGV(6,6),CVEGO(20,6),CVEGVO(6),AVEGV(6 ),AVEGO(20     0300
     1),AVEGVO,ABIOMA,CBIOMA(6),ALITT,CLITT(6),CORGT(6)    ,ABIOSP(20),     0320
     2TOT(6),AORGT,POPSP(20),ANIM(20,6),CPACTT(6),ABACT(3),ABACTT,           0340
     3 AVEG(20,6),ABIOM(98),ALIT(5),AORG(5),AMIN                            0360
C-----------------------------------------------------------------------
C      COMMON BLOCK /STAT/ CONTAINS THE STATE VARIABLES, AND /CHANGE/
C      THEIR INCREMENTS OR DECREMENTS FOR THE CURRENT TIME UNIT.
C-----------------------------------------------------------------------
       COMMON/STAT/ CVEG(20,6,6),CORG(5,6),POP(98),CBIOM(98,6),AQUA(6),     0380
     1 CLIT(5,6),CBACT(3,6),          DUMMY(96)
       COMMON /CHANGE/ CVEGQQ(20,6,6),CORGQQ(5,6),POPQQQ(98),CBIOMQ(98,6)   047
     1,AQUAQQ(6), CLITQQ(5,6),CPACTQ(3,6)           ,DUMMQ(96)
C-----------------------------------------------------------------------
C      COMMON BLOCK /DIAGR/ CONTAINS INFORMATION REQUIRED FOR GRAPHS.
C-----------------------------------------------------------------------
       COMMON /DIAGR/FIG(8,70), EXPLA(5,8),  TITLE(20),  YTITLE(10),
     1            XDOT(80),        XMAX,XMIN,YMAX,YMIN,NOSYM,NODOT,NOCOL    0480
     2 ,IYR, INITYP
C-----------------------------------------------------------------------
C      COMMON BLOCK /METEOR/ CONTAINS THE VALUES OF EXOGENOUS VARIABLES
C      FOR THE CURRENT TIME UNIT.
C-----------------------------------------------------------------------
       COMMON/METEOR/WIRRIG,ERO,RUNSOL(6),RUNDEB(3,6),DARAIN,DAYRUN,        0500
     1EVAP,WATTEM,DAPHOT,DAYRAD,DADUST(3,6),EXOG(98),RAINCO(6),            0520
     2COMPIN(6),DETIN(5,6),RUNON,FLOWIN,DRIFTV(20,6,6),DRIFTA(98,6),        0540
     3DRIFTM(3,6),DRIFPO(98)                                                0560
C-----------------------------------------------------------------------
C      COMMON BLOCK /PARAM/ CONTAINS THE VALUES OF PARAMETERS USED BY THE
C      PROCESS SUBROUTINES.
C-----------------------------------------------------------------------
       COMMON /PARAM/ P(10200)
       EQUIVALENCE (STNG,AGAIN),(CHNG,AGAINQ)                               0660
       EQUIVALENCE (STATE, CVEG), (DECINC, CVEGQQ),(SUMS,CVEGV)             0680
       DATA DECJAN/'JAN','FEB','MAR','APR','MAY','JUNE','JULY','AUG',   MAIN0420
     1 'SEPT','OCT','NOV','DEC'/,ZERO/'ZERO'/,ORIGIN/'0*' '/
       DATA CFRAC/' ','CARB','ON F','RACT','IONS'/                          0720
       DATA BICARB/'BICA'/, BICAR1/'RBON'/, BICAR2/'ATE '/
       DATA BLANK/' '/,NREPET/20*1/
       DATA/AMICRO/.0000001/,NOCOL/70/,HIGH/1.E20/,LIMIT/1586/              0820
       DATA LIMTOT/615/,LIMACC/21/                                          0840
       DATA MONDAY/31,28,31,30,31,30,31,31,30,31,30,31/                     0860
       DATA NYRDAY/365/,NREPET/20*1/,IRUN/1/                                0880
       TIME = EXTIME(Q)                                                     0900
       NCHAN = 2                                                            0920
       JDAY = 0                                                             0940
       JY = 2                                                               0960
    10 FORMAT (/////)                                                       0980
       WRITE (6,10)                                                     MAIN0500
    20 READ (5,250) (STATE(I), I=1,20)                                  MAIN0510
       WRITE(6,260) (STATE(I),I = 1,20)
       DO 40 I = 1, 20                                                  MAIN0540
       IF (STATE(I).NE.BLANK) GO TO 20                                  MAIN0550
    40 CONTINUE                                                         MAIN0560
```

```
      DO 50 I = 1, LIMIT                                              MAIN0570
   50 DECINC(I) = 0.                                                  MAIN0590
      DO 60 I=1,LIMACC                                                MAIN0600
      CING(I)=0.                                                      MAIN0610
   60 CHNG(I) = 0.                                                    MAIN0620
      DECMM = 0.                                                      MAIN0630
      NCHECK = 0                                                      MAIN0640
C-----------------------------------------------------------------------
C     A HEADING IS READ IN FOR TABULAR OUTPUT
C-----------------------------------------------------------------------
      READ(5,250)  (PLACE(I), I=1,20)                                 MAIN0650
C-----------------------------------------------------------------------
C     SPECIFICATIONS, SWITCHES AND INSTRUCTIONS ARE READ IN
C-----------------------------------------------------------------------

C..........N.B. NELEM, THE NUMBER OF CONSTITUENT ELEMENTS MODELLED, IS
C..........ASSUMED TO INCLUDE CARBON UNLESS TWO OR MORE CARBON
C..........FRACTIONS (THE NUMBER OF WHICH IS GIVEN IN NFRACT) ARE
C..........DISTINGUISHED.
      READ (5,230) NSPECV,NSPECA,NORGAN,NFRACT,NELEM,NOLIT,NCOHOR,
     1                      MICPOB,NUNIT,NOINST,NOTIME
      READ (5,230) JDAY, IYR, NDAY, NREP, NOGRAF, NOHIST
      TDAY = JDAY
      IF (NREP.GT.0) GO TO 63
      NREP = 1
      MREP(1) = NDAY
      GO TO 66
   63 READ (5,230) (MREP(I), I = 1, NREP)
   66 NELEMS = NELEM
      READ(5,230) MDEBUG,LDEBUG,NOSECS,ISENSE,NOREP,JSTATE,IPARAM,JPARAM
     1 ,NEWBEC,KDUMP,(IDUMP(I), I=1,KDUMP)
      IF ((MDEBUG.GT.0).AND.(MDEBUG.LE.JDAY)) NDEBUG = 1
      IF (LDEBUG.LT.MDEBUG) LDEBUG = NDAY
      IF(NFRACT.LE.0)  NELEMS = NELEM- 1                              MAIN0730
      IF (NUNIT.LE.0) NUNIT = 1000                                    MAIN0740
      UNIT = FLOAT(NUNIT) * .001                                      MAIN0750
      IF (NFRACT.LE.1) NFRACT = 0                                     MAIN0760
      IF (NORGAN.LE.0)  NORGAN= 1                                     MAIN0790

C..........INSTRUCTIONS TO PROCESS SUBROUTINES ARE PROVIDED
      IF (NOINST.GT.0) READ(5,230) (INSTRU(I),I=1,NOINST)             MAIN0800
      IF (NOTIME.LE.0) GO TO 80                                       MAIN0810

C..........FREQUENCY OF REPETITION OF PROCESS SUBROUTINES WITHIN A
C..........TIME UNIT IS SPECIFIED
      DO 70 I = 1, NOTIME                                             MAIN0820
   70 READ (5,230) J, NREPET(J)                                       MAIN0830
   80 IF (NFRACT.LE.0) NFRAC1=NELEM+1
      IF (NFRACT.GT.1) NFRAC1=NFRACT+NELEM+1
      VEGINC = UNIT/FLOAT(NREPET(1)) - .00001                         MAIN0850
      ANIINC = UNIT/FLOAT(NREPET(2)) - .00001                         MAIN0860
      AQUINC = UNIT/FLOAT(NREPET(3)) - .00001                         MAIN0870
      NFRELM=NFRACT+NELEM                                             MAIN0880
      NFRAC2=NFRACT+2                                                 MAIN0890
  150 IF (NSPECA.LE.0) GO TO 270                                      MAIN1150
C-----------------------------------------------------------------------
C     THE STAGES OF DEVELOPMENT FOR THE DIFFERENT ANIMAL SPECIES
```

```
C     GROUPS ARE SPECIFIED.
C-------------------------------------------------------------------------
      IF (NCOHOR.GT.1) GO TO 170                                        MAIN1160
      DO160 I = 1, NSPECA                                               MAIN1170
      NCOHCU(I) = I                                                     MAIN1180
  160 NCOH(I) = 1                                                       MAIN1190
      GO TO 270                                                         MAIN1200

C..........THE NUMBER OF STAGES OF DEVELOPMENT FOR EACH ANIMAL GROUP
C..........ARE READ.
  170 READ(5,230)(NCOH(I),I=1,NSPECA)                                   MAIN1210
      NCOHCU(1) = NCOH(1)                                               MAIN1220
      K1=0                                                              MAIN1230
      DO 220 I = 1, NSPECA                                              MAIN1240
      J=NCOH(I)                                                         MAIN1250
      IF (I.GT.1) NCOHCU(I) = NCOHCU(I-1) + J                           MAIN1260
      IF(J.LE.0) J=1                                                    MAIN1270
      IF(J.LE.NCOHOR)GO TO 190                                          MAIN1280
      WRITE (6,180)I,NCOHOR                                             MAIN1290
  180 FORMAT('0NUMBER OF COHORTS FOR SPECIES',I3,'EXCEEDS',I3)          MAIN1300
      STOP                                                              MAIN1310
  190 K=K1+1                                                            MAIN1340
      K1=K1+J                                                           MAIN1350
      IF(J.LE.1) GO TO 220                                              MAIN1360
      IF (J.LT.NCOHOR)GO TO 210                                         MAIN1370

C..........IF THE NUMBER OF STAGES OF DEVELOPMENT FOR THIS SPECIES
C..........GROUP IS MORE THAN ONE AND LESS THAN THE MAXIMUM, THEY ARE
C..........SPECIFIED.
      DO 200 L=K,K1                                                     MAIN1380
  200 LTSCOH(L)=L-K+1                                                   MAIN1390
      GO TO 220                                                         MAIN1400
  210 READ(5,230)(LTSCOH(L),L=K,K1)                                     MAIN1410
  220 CONTINUE                                                          MAIN1420
  230 FORMAT (16I5)                                                     MAIN1430
  240 FORMAT (8F10.2)                                                   MAIN1440
  250 FORMAT (20A4)                                                     MAIN1450
  260 FORMAT (1X, 30A4)                                                 MAIN1460
C-------------------------------------------------------------------------
C     THE NAMES OF THE VARIOUS ECOSYSTEM COMPONENTS ARE READ IN.
C-------------------------------------------------------------------------
  270 IF(NSPECV.GT.0)READ(5,280)((VSPNAM(I,J),J=1,7),I=1,NSPECV)        MAIN1470
  280 FORMAT (14A4)                                                     MAIN1480
      IF(NSPECA.GT.0)READ(5,280)((ASPNAM(I,J),J=1,7),I=1,NSPECA)        MAIN1490
      IF (NORGAN.LE.1) GO TO 300                                        MAIN1500
      READ(5,290)((ORGNAM(I,J), J = 1, 6), I=1,NORGAN)                  MAIN1510
  290 FORMAT (18A4)                                                     MAIN1520

C..........N.B. IF CARBON FRACTIONS ARE NOT DISTINGIISHED, TOTAL
C          CARBON IS ASSUMED TO HAVE THE FIRST PLACE IN THE LIST OF
C..........CONSTITUENTS.
  300 READ (5,250)((FRANAM(I,J), J = 1,3), I=1,NFRELM)                  MAIN1560
      IF  (NCOHOR.GT.1) READ(5,250) ((COHNAM(I,J),J=1,4),I=1,NCOHOR)    MAIN1580
      IF(NOLIT.GT.0)READ(5,250)((ALINAM(I,J),J=1,4),I=1,NOLIT)
      IF(MICROB.GT.0.0)
     1READ(5,250)((BACNAM(I,J),J=1,4),I=1,MICROB)
C-------------------------------------------------------------------------
```

```
C      INITIAL VALUES FOR THE STATE VARIABLES ARE READ IN, FROM AN
C      INPUT FILE IF THE SWITCH NEWBEG IS POSITIVE, OTHERWISE FROM CARD
C      INPUT.
C-----------------------------------------------------------------------
       IF  (NEWBEG.LE.0) GO TO 330
       REWIND 9
       READ (9) STATE
       GO TO 420
   330 IF (NSPECV.LE.0) GO TO 360                                       MAIN1690

C..........PLANT CONSTITUENTS
       DO 340 I = 1, NSPECV
       DO 340 J = 1, NORGAN                                             MAIN1720
       READ (5,240) (CVEC(I,J,K), K = 1, NFRELM)                        MAIN1730
   340 CONTINUE                                                         MAIN1740

   360 IF (NSPECA.LE.0) GO TO 380                                       MAIN1780
       DO 370 K = 1, NSPECA                                             MAIN1790
       K1=1                                                             MAIN1800
       IF(K.GT.1) K1=NCOHCU(K-1)+1                                      MAIN1810
       K2=NCOHCU(K)                                                     MAIN1820

C..........ANIMAL POPULATIONS
       READ(5,240) (POP(J), J=K1,K2)                                    MAIN1830
       READ(5,240)(BIOMIN(J),J=K1,K2)
       DO370J=K1,K2                                                     MAIN1840

C..........ANIMAL CONSTITUENTS
   370 READ (5,240)(CBIOM(J,I), I = 1, NFRELM)                          MAIN1850
       NSPCOH=NCOHCU(NSPECA)                                            MAIN1860
   380 IF(MICROB.LE.0.) GO TO 385
       DO 382 I=1,MICROB
   382 READ(5,240)(CBACT(I,K),K=1,NFRELM)
   385 IF (NOLIT.LE.0) GO TO 400                                        MAIN1870

C..........CONSTITUENTS OF DEAD MATERIAL
       DO 390 I = 1, NOLIT                                              MAIN1880
   390 READ (5,240) (CLIT(I,K), K = 1, NFRELM)
       DO 395 I=1,NOLIT
   395 READ (5,240) (CORG(I,K), K = 1, NFRELM)
   400 READ (5,240) (AQUA(K), K = 1, NFRAC1)
C-----------------------------------------------------------------------
C      THE STATE VARIABLES ARE TOTALLED
C-----------------------------------------------------------------------

C..........THE TOTAL ARRAYS ARE INITIALIZED
   420 IF (LOOP.LE.1) GO TO 428
       LOOP1 = LOOP - 1
       DO 425 I = 1, LOOP1
       IMIN = NEGATE(I)
       STATE(IMIN) = -STATE(IMIN) - AMICRO
   425 IF (ABS(STATE(IMIN)).LE.AMICRO) STATE(IMIN) = 0.
       IMIN = 0
   428 DO 430 I = 1, LIMTOT
   430 SUMS(I) = 0.                                                     MAIN1950
   435 IF (NSPECV.LE.0) GO TO 520
```

```
C..........PLANT STATE VARIABLES ARE TOTALLED
      DO 460 I = 1, NSPECV                                                MAIN1970
      DO 460 J = 1, NORGAN                                                MAIN1980
      IF (NFRACT.GT.0) GO TO 439                                              3080
      IF(CVEG(I,J,NFRELM).GT.0.) AVEG(I,J) = CVEG(I,J,NFRELM)
      GO TO 460
  439 AVEG(I,J)=0.
  440 DO 450 K1 = 1, NFRACT                                               MAIN2020
      K = K1+NELEM                                                        MAIN2030
      A = CVEG(I,J,K)                                                     MAIN2040
      IF (A.LE.0.) GO TO 450
      AVEGV(J) = AVEGV(J) + A                                             MAIN2050
      AVEG(I,J) = AVEG(I,J) + A                                           MAIN2060
  450 CONTINUE                                                            MAIN2070
  460 CONTINUE                                                            MAIN2080
  470 DO 490 K = 1, NFRELM                                                MAIN2090
      DO 490 I = 1, NSPECV
      DO 480 J = 1, NORGAN                                                MAIN2110
      A = CVEG(I,J,K)
      IF (A.LE.0.) GO TO 480
      CVEGO(I,K) = CVEGO(I,K) + A
  480 CONTINUE
  490 CVEGVO (K) = CVEGVO (K) + CVEGO (I,K)                               MAIN2130
      DO 510 I = 1, NSPECV
      DO 500 J = 1, NORGAN                                                MAIN2150
  500 AVEGO(I) = AVEGO(I) + AVEG(I,J)                                     MAIN2160
  510 AVEGVO = AVEGVO + AVEGO(I)                                          MAIN2170
  520 DO 550 I = 1, NOLIT
      AORG(I)=0.                                                              3500
      ALIT(I) = 0.                                                            3520
      DO 540 K1 = 1, NFRELM
      IF (K1.LE.NELEM) GO TO 530
      IF (CORG(I,K1).GT.0.)AORG(I) = AORG(I) + CORG(I,K1)
      IF (CLIT(I,K1).GT.0.) ALIT(I) = ALIT(I) + CLIT(I,K1)
  530 IF (CORG(I,K1).GT.0.)CORGT(K1) = CORGT(K1) + CORG(I,K1)
  540 IF (CLIT(I,K1).GT.0.) CLITT(K1) = CLITT(K1) + CLIT(I,K1)
      IF ((AORG(I).LE.0.).AND.(CORG(I,NFRELM).GT.0.))AORG(I) =
     1 CORG(I,NFRELM)
      IF ((ALIT(I).LE.0.).AND.(CLIT(I,NFRELM).GT.0.))ALIT(I) =
     1 CLIT(I,NFRELM)
      AORGT=AORGT+AORG(I)                                                     3680
  550 ALITT = ALITT + ALIT(I)
      IF (NSPECV.LE.0) GO TO 590
      DO 580 J=1,NORGAN
      DO 570 K = 1, NFRELM
      DO 560 I = 1, NSPECV
  560 IF (CVEG(I,J,K).GT.0.) CVEGV(J,K) = CVEGV(J,K) + CVEG(I,J,K)
  570 CONTINUE
  580 CONTINUE
  590 IF (MICROB.LE.0) GO TO 700
      ABACTT=0.                                                               3880
      DO 610 I=1,MICROB
      ABACT(I)=0.                                                             3920
      DO 600 K=1,NFRELM
      A=CBACT(I,K)                                                            3960
      IF (A.LE.0.) GO TO 600
      IF (K.GT.NELEM) ABACT(I)=ABACT(I)+A                                     3980
```

```
      CBACTT(K) = CBACTT(K) + A
  600 CONTINUE
      IF ((NFRACT.LE.0).AND.(CBACT(I,NELEM).GT.0.))ABACT(I) =
     1 CBACT(I,NELEM)
  610 ABACTT=ABACTT+ABACT(I)
  700 IF (NSPECA.LE.0) GO TO 740                                   MAIN2790

C..........ANIMAL STATE VARIABLES ARE TOTALLED
      ABIOMA = 0.                                                  MAIN2800
      DO 730 I = 1, NSPECA                                         MAIN2810
      K1=1                                                         MAIN2820
      IF(I.GT.1)K1 = NCOHCU(I-1)+1                                 MAIN2830
      K2 = NCOHCU(I)                                               MAIN2840
      DO 730 J = K1,K2                                             MAIN2850
  707 ABIOM(J) = 0.
      DO 720 K = 1, NFRELM                                         MAIN2870
      A = CBIOM(J,K)                                               MAIN2880
      IF (A.LE.0.) GO TO 720
      IF (K.LE.NELEM) GO TO 710                                    MAIN2890
      ABIOM(J)  = ABIOM(J) + A                                     MAIN2900
      ABIOMA = ABIOMA + A                                          MAIN2910
      ABIOSP(I) = ABIOSP(I) + A                                    MAIN2920
  710 ANIM(I,K)=ANIM(I,K)+A                                        MAIN2930
      CBIOMA(K) = CBIOMA(K) + A
  720 CONTINUE
      IF ((ABIOM(J).LE.0.).AND.(CBIOM(J,NFRELM).GT.0.))ABIOM(J) =
     1  CBIOM(J,NFRELM)
      IF (POP(J).GT.0.) POPSP(I) = POPSP(I) + POP(J)
  730 CONTINUE                                                     MAIN2970
      IF (ABIOMA.LE.0.) ABIOMA = CBIOMA(NFRELM)                    MAIN2980
  740 CONTINUE
      DO750 K = 1,NELEM
  750 TOT(K) = CORGT(K)+ CLITT(K)
      ATOT = AORGT+ ALITT

C..........TOTALS FOR THE WHOLE SYSTEM ARE CALCULATED
  760 DO 770 K = 1, NELEM
  770 TOTAL(K) = CBIOMA(K) + CVEGVO(K) + CBACTT(K) + TOT(K)        MAIN3060
      ATOTO = ABIOMA + AVEGVO + ABACTT + ATOT
      IF (IMIN.GT.0) GO TO 1320                                    MAIN3080
      IF (NCHECK.NE.0) GO TO 1340                                  MAIN3090
C-----------------------------------------------------------------------
C     THE DATE IS INITIALIZED
C-----------------------------------------------------------------------
      INITYR = IYR                                                 MAIN3110
      IF (MOD(IYR,4).GT.0) GO TO 780                               MAIN3120
      MONDAY(2) = 29                                               MAIN3130
      NYRDAY = 366                                                 MAIN3140
  780 IF (JDAY.LE.0) JDAY = 1                                      MAIN3150
      IF (NDAY.LE.JDAY) NDAY = JDAY + 1                            MAIN3160
      K = 0                                                        MAIN3170
C-----------------------------------------------------------------------
C     GRAPHING INSTRUCTIONS ARE READ
C-----------------------------------------------------------------------
      NOHIS = NOGRAF+1                                             MAIN3180
      NOHISU = NOGRAF + NOHIST                                     MAIN3190
  790 IF (NOGRAF.LE.0) GO TO 860                                   MAIN3200
```

```
C..........INSTRUCTIONS FOR LINE GRAPHS
      DO 850 I = 1, NOGRAF                                              MAIN3210
      I1 = K + 1                                                        MAIN3220
      READ(5,230) (MGRA(J),J=1,8)
      DO 800 J = 1, 8
      IF (MGRA(J).LE.0) GO TO 810
      K = K + 1
      LIGRAF(K) = MGRA(J)
  800 CONTINUE                                                          MAIN3280
  810 LISTER(I) = K                                                     MAIN3290
      READ (5,250) (TITLES(I,J) , J = 1,20)                             MAIN3300
      READ (5,250) (YAXISS(I,J),J=1,10), ORIGIN(I)                      MAIN3310
      IF (K.LE.I1) GO TO 830                                            MAIN3320
      DO 820 J = I1,K                                                   MAIN3330
  820 READ (5,250) (EXPLAN(L,J), L = 1,5)                               MAIN3340
      GO TO 850                                                         MAIN3350
  830 DO 840 J = 1,5                                                    MAIN3360
  840 EXPLAN(J,I1) = BLANK                                              MAIN3370
  850 CONTINUE                                                          MAIN3380
  860 IF (NOHIST.LE.0) GO TO 890                                        MAIN3390

C..........INSTRUCTIONS FOR BLOCK GRAPHS
      K1 = K + 1                                                        MAIN3400
      K2 = K + NOHIST                                                   MAIN3410
      READ (5,230) (LIGRAF(I), I = K1, K2)                              MAIN3420
      DO 870 I = NOHIS,NOHISU                                           MAIN3430
      READ (5,250) (TITLES(I,J), J = 1, 20)                             MAIN3440
  870 READ (5,250) (YAXISS(I,J),J=1,10), ORIGIN(I)                      MAIN3450
  880 FORMAT (' ',20A4)                                                 MAIN3460
C-----------------------------------------------------------------------
C     SPECIFICATIONS OF TABULAR OUTPUT ARE INITIALIZED
C-----------------------------------------------------------------------
  890 J = 0                                                             MAIN3520
      DO 910 I = 1, NREP                                                MAIN3530
      IF ((MREP(I).LE.0).OR.(MREP(I).GT.NDAY)) GO TO 910                MAIN3540
      IF((I.GT.1).AND.(MREP(I).LE.MREP(I-1)))GO TO 910                  MAIN3550
      J = J + 1                                                         MAIN3560
      MREP(J) = MREP(I)                                                 MAIN3570
  910 CONTINUE                                                          MAIN3580
      NREP = J                                                          MAIN3590
      IF (J.EQ.0) GO TO 920                                             MAIN3600
      IF (MREP(J).EQ.NDAY) GO TO 930                                    MAIN3610
  920 NREP = NREP + 1                                                   MAIN3620
      MREP(NREP) = NDAY                                                 MAIN3630
  930 IREP = 1                                                          MAIN3640
      IF (NOHISU.LE.0) GO TO 940                                        MAIN3650
      XMIN = JDAY                                                       MAIN3660
      XMAX = NDAY                                                       MAIN3670
      PERIOD = (XMAX - XMIN)/69.                                        MAIN3680
  940 TDAY=JDAY                                                         MAIN3690
      IYRDAY = IDAY                                                     MAIN3700
C-----------------------------------------------------------------------
C     PROCESS SUBROUTINES ARE CALLED TO PERMIT PARAMETERS TO BE READ
C-----------------------------------------------------------------------
      CALL AINPUT
      CALL EINPUT
```

```
      CALL MINPUT
      CALL VINPUT
C-----------------------------------------------------------------------
C     IF REQUIRED, PART OF THE COMMON BLOCK /PARAM/ IS PRINTED OUT.
C-----------------------------------------------------------------------
      IF (IPARAM.LE.0) GO TO 947
      WRITE (6,943) IPARAM, JPARAM
  943 FORMAT('0COMMON BLOCK /PARAM/ FROM ADDRESS', I6,' TO ADDRESS',I6/)
      WRITE (6,945) (P(I), I = IPARAM, JPARAM)
  945 FORMAT (1X, 10G12.5)
  947 IRUN = 1
      IF (ISENSE.EQ.0) GO TO 960                                        MAIN3750
C-----------------------------------------------------------------------
C     IF SENSITIVITY TESTS ARE BEING PERFORMED, THE SUBROUTINE SENSIT
C     IS CALLED TO SET THE INITIAL CONDITIONS
C-----------------------------------------------------------------------
      CALL SENSIT(IRUN,1,NRUN)
  950 IF(IRUN.EQ.1) GO TO 960                                           MAIN3770
      IREP = 1                                                          MAIN3780
      CALL SENSIT(IRUN,NRA,NRUN)
      IDAY = JDAY                                                       MAIN3800
      IYRDAY = JDAY                                                     MAIN3810
      NCHECK = 0                                                        MAIN3820
  960 CONTINUE                                                          MAIN3830
      IF (NOHISU.LE.0) GO TO 980                                        MAIN3840
C-----------------------------------------------------------------------
C     LIMITS FOR THE GRAPHS ARE INITIALIZED
C-----------------------------------------------------------------------
      DO 970 I = 1, NOHISU                                              MAIN3850
      AMINI(I) = HIGH                                                   MAIN3860
  970 AMAXI(I) = -HIGH                                                  MAIN3870
  980 FRAC = 0.
C-----------------------------------------------------------------------
C     THE CALENDAR MONTH IS DETERMINED
C-----------------------------------------------------------------------
      MONEND = 0                                                        MAIN3900
      MONTH = 0                                                         MAIN3910
      NUMMON = JDAY                                                     MAIN3920
  990 MONTH = MONTH + 1                                                 MAIN3930
      MONEND = MONEND + MONDAY(MONTH)                                   MAIN3940
      IF (MONTH.GT.1) NUMMON = NUMMON - MONDAY(MONTH-1)                 MAIN3950
      IF ( IYRDAY.GT.MONEND) GO TO 990                                  MAIN3960
      IF (IRUN.GT.1) GO TO 1510                                         MAIN3970
C-----------------------------------------------------------------------
C     THE SUBROUTINE EXTERN IS CALLED TO RECEIVE INPUT OF
C     EXOGENOUS VARIABLES
C-----------------------------------------------------------------------
      CALL EXTERN
      IF (NOREP.GE.2) GO TO 1020
C-----------------------------------------------------------------------
C     A HEADING IS PRINTED FOR THE INITIAL REPORT
C-----------------------------------------------------------------------
      WRITE(6,1000)  (PLACE(I),I=1,20)                                  MAIN4000
 1000 FORMAT ('1', 20A4)                                                MAIN4010
      WRITE (6,1010) DECJAN(MONTH), NUMMON, IYR                         MAIN4020
 1010 FORMAT('0INITIAL REPORT ON ', A4, I3, I5)                         MAIN4030
 1020 IF (NOHISU.LE.0) GO TO 1140                                       MAIN4050
```

```
C-----------------------------------------------------------------------
C      THE GRAF SUBROUTINE IS SUPPLIED WITH CURRENT VALUES FOR THE
C      VARIABLES TO BE GRAPHED
C-----------------------------------------------------------------------
       I2 = 0
       IF (NOGRAF.LE.0) GO TO 1080                                      MAIN4060
       DO 1070 I = 1, NOGRAF                                            MAIN4070
       I1 = I2 + 1                                                      MAIN4080
       I2 = LISTER(I)                                                   MAIN4090
       DO 1070 J =I1, I2                                                MAIN4100
       I3 = LIGRAF(J)                                                   MAIN4110
       I4 = I3/10000                                                    MAIN4120
       I3 = I3 - I4*10000                                               MAIN4130
       I4 = I4 + 1                                                      MAIN4140
       GO TO (1050,1030,1040      ), I4                                 MAIN4150
  1030 A =SUMS(I3)
       GO TO 1060                                                       MAIN4170
  1040 A = STNG(I3)                                                     MAIN4180
       GO TO 1060                                                       MAIN4190
  1050 A = STATE (I3)                                                   MAIN4200
  1060 AMINI(I) = AMIN1(AMINI(I),A)                                     MAIN4210
       AMAXI(I) = AMAX1(AMAXI(I),A)                                     MAIN4220
  1070 FIGS (J,1) = A                                                   MAIN4230
  1080 IF (NOHIST.LE.0) GO TO 1490                                      MAIN4240
       I1= LISTER(NOGRAF)                                               MAIN4250
       I2 = I1 + NOHIST                                                 MAIN4260
       I1 = I1 + 1                                                      MAIN4270
       I = NOGRAF                                                       MAIN4280
       DO 1130 J =I1, I2                                                MAIN4290
       I = I + 1                                                        MAIN4300
       I3 = LIGRAF(J)                                                   MAIN4310
       I4 = I3/10000                                                    MAIN4320
       I3 = I3 - I4*10000                                               MAIN4330
       I4 = I4 + 1                                                      MAIN4340
       GO TO(1110,1090,1100      ), I4                                  MAIN4350
  1090 A = SUMS(I3)
       GO TO 1120                                                       MAIN4370
  1100 A = STNG(I3)                                                     MAIN4380
       GO TO 1120                                                       MAIN4390
  1110 A = STATE(I3)                                                    MAIN4400
  1120 AMINI(I) = A                                                     MAIN4410
       AMAXI(I) = A                                                     MAIN4420
  1130 FIGS(J,1) = A                                                    MAIN4430
  1140 IF (NOREP.GE.2) GO TO 1150                                       MAIN4440
       GO TO 1490                                                       MAIN4450
  1150 IF (NOSECS.LE.1) GO TO 1160                                      MAIN4460
C-----------------------------------------------------------------------MAIN4470
C      THE CPU TIMER IS REPORTED AND RE-SET
C-----------------------------------------------------------------------
       TIMER = EXTIME(Q)                                                MAIN4480
       WRITE (6,1220) TIMER                                             MAIN4490
C-----------------------------------------------------------------------
C      THE FOLLOWING SECTION ACTUALLY PERFORMS THE SIMULATION OVER A
C      SINGLE TIME UNIT
C-----------------------------------------------------------------------

C..........THE LIMITS FOR THE TIME UNIT OF SIMULATION ARE SET AND THE
```

```
C..........TIME-UNIT LOOP IS INITIALIZED
 1160 ORIG = FLOAT(IDAY) + FRAC                                          MAIN4500
      IDAY1 = IDAY                                                       MAIN4510
      IDAY2 = ORIG + UNIT                                                MAIN4520
      IYRDAZ = IYRDAY                                                    MAIN4530
      MONTHZ = MONTH                                                     MAIN4540
      NUMMOZ = NUMMON                                                    MAIN4550
      MONENZ = MONEND                                                    MAIN4560
      NYRDAZ = NYRDAY                                                    MAIN4570
      IYRZ = IYR                                                         MAIN4580
      LOOP = 0                                                           MAIN4590
      FACTO = 1.                                                         MAIN4600
      FACTOR = 0.                                                        MAIN4610

C..........INITIALIZATION FOR THE CURRENT REPETITION OF THE TIME-UNIT
C..........LOOP IS PERFORMED
 1170 VEGCO = 0.                                                         MAIN4620
      AQUACO=0.
      ANIMCO= 0.                                                         MAIN4640
      FACTOR = FACTO                                                     MAIN4650
      IDAY = ORIG                                                        MAIN4660
      LOOP = LOOP + 1                                                    MAIN4670
      IF (LOOP.LE.20) GO TO 1190                                         MAIN4680
      WRITE (6,1180) IDAY                                                MAIN4690
 1180 FORMAT (' TIME LOOP ATTEMPTED TWENTY TIMES  AT DAY', I4)           MAIN4700
      STOP                                                               MAIN4710
 1190 IMIN = 0                                                           MAIN4720
      IYRDAY = IYRDAZ                                                    MAIN4740
      MONTH = MONTHZ                                                     MAIN4750
      NUMMON = NUMMOZ                                                    MAIN4760
      MONEND = MONENZ                                                    MAIN4770
      NYRDAY = NYRDAZ                                                    MAIN4780
      IYR = IYRZ                                                         MAIN4790
      DO 1280 IDAY = IDAY1, IDAY2                                        MAIN4800
      IF (IDAY.LE.IDAY1) GO TO 1230                                      MAIN4810
      NUMMON = NUMMON + 1                                                MAIN4820
      IYRDAY = IYRDAY + 1                                                MAIN4830
      IF (IYRDAY.LE.NYRDAY) GO TO 1200                                   MAIN4840
      MONTH = 1                                                          MAIN4850
      MONEND = MONDAY(1)                                                 MAIN4860
      IYRDAY = IYRDAY - NYRDAY                                           MAIN4870
      NUMMON = IYRDAY                                                    MAIN4880
      IYR = IYR + 1                                                      MAIN4890
      NYRDAY = 365                                                       MAIN4900
      MONDAY(2) = 28                                                     MAIN4910
      IF (MOD(IYR,4).GT.0) GO TO 1200
      MONDAY(2) = 29                                                     MAIN4930
      NYRDAY = 366
 1200 IF (IYRDAY.LE.MONEND) GO TO 1210                                   MAIN4950
      NUMMON = IYRDAY - MONEND
      MONTH = MONTH + 1                                                  MAIN4970
      MONEND = MONEND + MONDAY(MONTH)                                    MAIN4980

C..........THE SUBROUTINE EXOGEN IS CALLED FOR CURRENT VALUES OF
C..........THE EXOGENOUS VARIABLES
 1210 CALL EXOGE2
 1220 FORMAT ( 104X, F10.3, ' SECONDS ELAPSED')                          MAIN5000
```

```
      IF (LOOP.LE.1) PRECMM = PRECMM + DARAIN                           MAIN5010
 1230 DAYDAY = AMIN1 (UNIT, FLOAT(IDAY-IDAY1+1))                        MAIN5020

C..........THE PROCESS SUBROUTINES ARE CALLED AS FREQUENTLY
C..........AS NECESSARY WITHIN EACH DAY OF THE TIME UNIT.
      IF (NSPECV.LE.0) GO TO 1250                                       MAIN5030
 1240 IF ((VEGCO + VEGINC).GT.DAYDAY) GO TO 1250                        MAIN5040
      VEGCO = VEGCO + VEGINC                                            MAIN5050
      CALL VEGET                                                        MAIN5060
      GO TO 1240                                                        MAIN5070
 1250 IF (NSPECA.LE.0) GO TO 1270                                       MAIN5080
 1260 IF ((ANIMCO + ANIINC).GT.DAYDAY) GO TO 1270                       MAIN5090
      ANIMCO = ANIMCO + ANIINC                                          MAIN5100
      CALL ANIMAL                                                       MAIN5110
      GO TO   1260                                                      MAIN5120
 1270 IF((AQUACO + AQUINC).GT.DAYDAY)  GO TO 1280
      AQUACO = AQUACO + AQUINC
      CALL MEDIUM
      GO TO 1270                                                        MAIN5160
 1280 NCHECK = 1                                                        MAIN5170

C..........PROPOSED DECREMENTS ARE TESTED TO ENSURE THAT STATE
C..........VARIABLES ARE ADEQUATE TO MEET THEM.  OTHERWISE, THE
C..........TIME UNIT IS REDUCED
      DO 1290 I = 1, LIMIT                                              MAIN5180
      IF((STATE(I).LE.0.).OR.(DECINC(I)+STATE(I)).GE.0.) GO TO 1290
      A =      - STATE(I)/DECINC(I)                                     MAIN5200
      IF (A.GE.FACTOR) GO TO 1290                                       MAIN5210
      FACTOR = A                                                        MAIN5220
      IMIN = I                                                          MAIN5230
 1290 CONTINUE                                                          MAIN5240

C..........INCREMENTS ARE APPLIED, TO THE STATE VARIABLES AND
C..........ACCUMULATORS, AND THE INCREMENT ARRAYS ARE RE-INITIALIZED
      DO 1300 I = 1, LIMIT                                              MAIN5250
      IF (DECINC(I).EQ.0.) GO TO 1300                                   MAIN5260
      A = DECINC(I)                                                     MAIN5270
      IF (A.EQ.0.) GO TO 1300
      IF (STATE(I).LT.0.) A =  -A
      IF (FACTOR.LT.1.) A = A * FACTOR                                  MAIN5280
      STATE(I) = STATE(I) + A                                           MAIN5290
      IF (ABS(STATE(I)).LT.AMICRO) STATE(I) = 0.
      DECINC(I) = 0.                                                    MAIN5310
 1300 CONTINUE                                                          MAIN5320
      DO 1310 I = 1, LIMACC                                             MAIN5330
      STNG (I) = STNG(I) + CHNG(I) * FACTOR                             MAIN5340
 1310 CHNG(I) = 0.                                                      MAIN5350
      IF (IMIN) 420,420,428

C..........IF ANY STATE VARIABLES HAVE BEEN INADEQUATE TO MEET
C..........THE PROPOSED DECREMENTS, THE TIME-UNIT LOOP IS RE-ENTERED
 1320 FACTO = FACTO - FACTOR                                            MAIN5370
      STATE(IMIN) = - AMICRO
      NEGATE(LOOP) = IMIN
      WRITE (6,1330) IMIN, FACTOR, IDAY, FPAC                           MAIN5380
 1330 FORMAT(' STATE(',I5,') PERMITS ONLY',F13.10,' OF THE PROPOSED UNITMAIN5390
     1 CHANGE AT',I4,' + ',F5.3,' DAYS')                                MAIN5400
```

```
      GO TO 1170                                                    MAIN5410
 1340 IDAY = IDAY2                                                  MAIN5440
      FRAC = AMOD ((UNIT+FRAC),1.)                                  MAIN5450
      IF (IDAY.LE.IDAY1) GO TO 1150                                 MAIN5460
      IF (NOHISU.LE.0) GO TO 1450                                   MAIN5470
C----------------------------------------------------------------------------
C     IF GRAPHS ARE REQUIRED, THE CURRENT VALUES OF VARIABLES
C     FOR GRAPHING ARE RECORDED
C----------------------------------------------------------------------------
      JX = (FRAC + FLOAT(IDAY-JDAY))/PERIOD + 1.                    MAIN5480
      IF (IDAY.EQ.NDAY) JX = 70                                     MAIN5490
      IF (JX.EQ.JXXX) GO TO 1450                                    MAIN5500
      JXXX = JX                                                     MAIN5510
      I2 = 0                                                        MAIN5520
      I = 0                                                         MAIN5530
 1360 I = I + 1                                                     MAIN5540
      IF (I.GT.NOHISU) GO TO 1440                                   MAIN5550
      I1 = I2 + 1                                                   MAIN5560
      I2 = I1                                                       MAIN5570
      IF (I.LE.NOGRAF) I2 = LISTER(I)                               MAIN5580
      DO 1430 J = I1, I2                                            MAIN5590
      I3 = LIGRAF(J)                                                MAIN5600
      I4 = I3/10000                                                 MAIN5610
      I3 = I3 - I4*10000                                            MAIN5620
      I4 = I4 + 1                                                   MAIN5630
      GO TO (1390,1370,1380     ), I4
 1370 A =SUMS(I3)                                                   MAIN5650
      GO TO 1400                                                    MAIN5660
 1380 A = STNG(I3)                                                  MAIN5670
      GO TO 1400                                                    MAIN5680
 1390 A = STATE (I3)                                                MAIN5690
 1400 FIGS(J,JX) = A                                                MAIN5700
      IF (JX.LE.JY) GO TO 1420                                      MAIN5710
      JX1 = JX - 1                                                  MAIN5720
      ADD = (    A      - FIGS(J,JY-1))/FLOAT(JX - JY + 1)          MAIN5730
      BADD = FIGS(J,JY-1)                                           MAIN5740
      DO 1410 K = JY, JX1                                           MAIN5750
      BADD = BADD + ADD                                             MAIN5760
 1410 FIGS(J,K) =              BADD                                 MAIN5770
 1420 CONTINUE                                                      MAIN5780
      AMAXI(I) = AMAX1(AMAXI(I), A)                                 MAIN5790
 1430 AMINI(I) = AMIN1(AMINI(I), A)                                 MAIN5800
      GO TO 1360                                                    MAIN5810
 1440 JY = JX + 1                                                   MAIN5820
 1450 IF (MDEBUG.LE.0) GO TO 1455
      NDEBUG = 0
      IF ((IDAY.GE.MDEBUG).AND.(IDAY.LE.LDEBUG)) NDEBUG = 1
C----------------------------------------------------------------------------
C     STATE VARIABLES MAY BE DUMPED ON LOGICAL UNITS 10 ETC. IF NEEDED.
C----------------------------------------------------------------------------
 1455 IF (KDUMP.LE.0) GO TO 1458
      IF (IDUMP(JDUMP).GT.IDAY) GO TO 1458
      REWIND IOUNIT
      WRITE (IOUNIT) STATE
      END FILE IOUNIT
      WRITE (6,1456) IDAY, IOUNIT
 1456 FORMAT ('1STATE VARIABLES DUMPED AT DAY', I5,' ON UNIT', I3)
```

```
      JDUMP = JDUMP + 1
      IF (JDUMP.GT.KDUMP) KDUMP = 0
      IOUNIT = IOUNIT + 1
 1458 IF (ISENSE.EQ.0) GO TO 1460
C-----------------------------------------------------------------------
C     IF SENSITIVITY TESTS ARE TO BE PERFORMED, THE CURRENT VALUES
C     OF THE VARIABLES REQUIRED ARE RECORDED
C-----------------------------------------------------------------------
      ISW = 1
      CALL SENOUT (ISW,IDAY,IRUN)                                       MAIN5840
 1460 IF ((IDAY.LT.MREF(IREP)).AND.(IDAY.LT.NDAY)) GO TO 1150           MAIN5850
C-----------------------------------------------------------------------MAIN5860
C     IF A TABULAR REPORT IS REQUIRED AT THIS STAGE OF THE
C     SIMULATION, IT IS PRODUCED.
C-----------------------------------------------------------------------
      IF ((NOREP.EQ.1).OR.(NOREP.EQ.2)) GO TO 1510                      MAIN5870
      WRITE (6,1300) (PLACE(I), I=1,20)                                 MAIN5880
      KDAY = IDAY - JDAY                                                MAIN5890
      WRITE(6,1470) IREP,DECJAN(MONTH),NUMMON,IYR,KDAY                  MAIN5900
 1470 FORMAT ('0REPORT NO.', I3, ' ON ', A4, I3, I5,' (I.E., AFTER',I4,
     1' DAYS OF SIMULATION)')
      IF (FRAC.GT.0.0005) WRITE (6,1480) FRAC                           MAIN5930
 1480 FORMAT ('+', 64X,' + ',F5.3,' DAY')                               MAIN5940
      IREP = IREP + 1                                                   MAIN5950
 1490 CALL REPORT                                                       MAIN5960
      IF (NOSECS.LE.0) GO TO 1510                                       MAIN5970
C-----------------------------------------------------------------------
C     THE CPU TIMER IS REPORTED AND RE-SET, AND
C     THE SIMULATION IS CONTINUED UNLESS COMPLETE
C-----------------------------------------------------------------------
 1500 FORMAT    ('+',107X,F10.3,' SECONDS ELAPSED')                     MAIN5980
      TIMER= EXTIME(0)                                                  MAIN5990
      WRITE (6,1500) TIMER                                              MAIN6000
 1510 IF (IDAY.LT.NDAY) GO TO 1150                                      MAIN6010
C-----------------------------------------------------------------------
C     IF SIMULATION IS COMPLETE, ANY GRAPHS REQUIRED ARE PRINTED
C-----------------------------------------------------------------------
      I2 = 0                                                            MAIN6020
      IF (NOGRAF.LE.0) GO TO 1800                                       MAIN6030

C..........LINE GRAPHS
      DO 1590 I = 1, NOGRAF                                             MAIN6040
      I1 = I2 + 1                                                       MAIN6050
      I2 = LISTER(I)                                                    MAIN6060
      I3 = 0                                                            MAIN6070
      DO 1560 K = I1, I2                                                MAIN6080
      I3 = I3 + 1                                                       MAIN6090
      DO 1520 J = 1,5                                                   MAIN6100
 1520 EXPLA(J,I3) = EXPLAN(J,K)                                         MAIN6110
      YMAX = AMAXI(I)                                                   MAIN6120
      YMIN = AMINI(I)                                                   MAIN6130
      IF((ORIGIN(I).NE.ZERO).OR.((YMAX.GT.0.).AND.(YMIN.LT.0.)))GO TO   MAIN6140
     1 1550                                                             MAIN6150
      IF (YMAX) 1530,1550,1540                                          MAIN6160
 1530 YMAX = 0.                                                         MAIN6170
      GO TO 1550                                                        MAIN6180
 1540 YMIN = 0.                                                         MAIN6190
```

```
 1550 DO 1560 J = 1, 70                                                MAIN6200
 1560 FIG(I3,J) = FIGS(K,J)                                            MAIN6210
      DO 1570 J = 1, 20                                                MAIN6220
 1570 TITLE(J) = TITLES(I,J)                                           MAIN6230
      DO 1580 J = 1, 10                                                MAIN6240
 1580 YTITLE(J) = YAXISS(I,J)                                          MAIN6250
      NOSYM = I7                                                       MAIN6260
      CALL GRAF                                                        MAIN6270
      IF (NOSECS.LE.0) GO TO 1590                                      MAIN6280
      TIMER=  EXTIME(Q)                                                MAIN6290
      WRITE(6,1500)TIMER                                               MAIN6300
 1590 CONTINUE                                                         MAIN6310
 1600 IF (NOHISH.LE.NOGRAF) GO TO 1690                                 MAIN6320

C..........BLOCK GRAPHS
      I1 = I2 + NOHIST                                                 MAIN6330
      I2 = I2 + 1                                                      MAIN6340
      K = NOGRAF                                                       MAIN6350
 1610 DO 1680 I = I2, I1                                               MAIN6360
      K = K + 1                                                        MAIN6370
      YMAX = AMAXI(K)                                                  MAIN6380
      YMIN = AMINI(K)                                                  MAIN6390
      IF((ORIGIN(I).NE.ZERO).OR.((YMAX.GT.0.).AND.(YMIN.LT.0.)))GO TO  MAIN6400
     1 1640                                                            MAIN6410
     IF (YMAX) 1620,1640,1630                                          MAIN6420
 1620 YMAX = 0.                                                        MAIN6430
      GO TO 1640                                                       MAIN6440
 1630 YMIN = 0.                                                        MAIN6450
 1640 DO 1650 J = 1, 70                                                MAIN6460
 1650 XDOT(J) = FIGS(I,J)                                              MAIN6470
      DO1660 J = 1, 20                                                 MAIN6480
 1660 TITLE(J) = TITLES(K,J)                                           MAIN6490
      DO 1670 J = 1,10                                                 MAIN6500
 1670 YTITLE(J) = YAXISS(K,J)                                          MAIN6510
      CALL HIST                                                        MAIN6520
C-----------------------------------------------------------------------
C     THE CPU TIMER IS REPORTED AND RE-SET
C-----------------------------------------------------------------------
      TIMER= EXTIME(Q)                                                 MAIN6530
      WRITE (6,1500) TIMER                                             MAIN6540
 1680 CONTINUE                                                         MAIN6550
 1690 CONTINUE                                                         MAIN6560
      IF (ISENSE.EQ.0) STOP                                            MAIN6570
C-----------------------------------------------------------------------
C     IF SENSITIVITY TESTS ARE REQUIRED, THE SUBROUTINE SENOUT
C     IS CALLED TO RECORD FINAL VALUES OF THE VARIABLES, AND, IF THE
C     LAST RUN HAS BEEN COMPLETED, TO PRINT OUT THE RESULTS.
C-----------------------------------------------------------------------
      ISW = 2                                                          MAIN6580
      CALL SENOUT (ISW,IDAY,IRUN)                                      MAIN6590
      IRUN = IRUN + 1                                                  MAIN6600
      IF (IRUN.LE.30) GO TO 950                                        MAIN6610
      STOP                                                             MAIN6620
      END                                                              MAIN6630
```

```
      SUBROUTINE EXTERN

C COMPTN           THE MEAN CONCENTRATION OF DISSOLVED CHEMICAL
C                  CONSTITUENTS IN THE INFLOWING WATER
C DAPHOT           PHOTOPERIOD FOR THE CURRENT DAY (HRS)
C DAPATN           AMOUNT OF RAINFALL ON THE CURRENT DAY
C DAYRAD           RADIATION FOR THE CURRENT DAY
C DAYRUN           THE AMOUNT OF RUNON ON THE CURRENT DAY
C DETTN (K,L)      CONTENT (IN G.PER G.) OF THE K'TH CONSTITUENT IN
C                  THE L'TH CATEGORY OF FLOATING DETRITUS IN THE
C                  INFLOWING WATER
C DRIFTA(L,N)      THE AMOUNT OF THE N'TH CONSTITUENT OF THE L'TH
C                  ANIMAL COHORT ENTERING PER UNIT OF THE ECOSYSTEM
C                  BY DRIFT IN A SINGLE TIME UNIT.
C DRIFTM(T,N)      THE AMOUNT OF THE N'TH CONSTITUENT IN THE T'TH
C                  TYPE OF MICRO-ORGANISM ENTERING PER UNIT AREA OF THE
C                  ECOSYSTEM BY DRIFT IN A SINGLE TIME UNIT
C DRIFTV(I,J,N)    THE AMOUNT OF THE N'TH CONSTITUENT OF THE I'TH
C                  PLANT SPECIES GROUP AND THE J'TH ORGAN TYPE
C                  ENTERING PER UNIT AREA OF THE ECOSYSTEM BY DRIFT IS
C ERO              THE QUANTITY OF ERODED SOIL IMPORTED ON THE
C                  PRESENT DAY
C EVAP             PAN EVAPORATION FOR THE CURRENT DAY
C LIMEX            THE FIRST WORD IN THE METEOR COMMON BLOCK REQUIRING
C                  REINITIALIZATION EVERY TIME EXTERN IS CALLED
C LIMEXC           THE LAST WORD IN THE METEOR COMMON BLOCK REQUIRING
C                  REINITIALIZATION EVERY TIME EXTERN IS CALLED
C NTRRTG           NUMBER OF DAYS ON WHICH WATER IS WITHDRAWN
C NRATN            NUMBER OF DAYS ON WHICH PRECIPITATION IS RECORDED
C NRUNON           NUMBER OF DAYS ON WHICH SURFACE FLOW IS RECORDED
C RATNCO           THE MEAN ELEMENTAL COMPOSITION OF PRECIPITATION
C RUNDEB           THE AMOUNT OF CHEMICAL CONSTITUENTS IN DETRITUS
C                  IMPORTED WITH SURFACE FLOW
C RUNSOL           THE AMOUNTS OF CHEMICAL ELEMENTS IN INORGANIC FORM
C                  IMPORTED WITH THE SURFACE FLOW
C WATTEM           WATER TEMPERATURE FOR THE CURRENT DAY
C WIRRIG           THE AMOUNT OF WATER WITHDRAWN ON THE CURRENT DAY
      DIMENSION MRAIN(100),RAIN(100),MRUNON(50),RUNON(50),
     1 ERODED(50),RUNMIN(50,6),RUNORG(50,3,6),
     2 EVAPOR(12),PHOTOP(12),RADIA(12),
     3 EXO(132),MIRRIG(50),WATIRR(50),WTEMP(12)
C-----------------------------------------------------------------------
C      COMMON BLOCK /SPEC/ CONTAINS SPECIFICATIONS AND OTHER INFORMATION
C      COMMON TO THE WHOLE SET OF PROGRAMS, BUT EXCLUDING STATE AND
C      EXOGENOUS VARIABLES.
C-----------------------------------------------------------------------
      COMMON/SPEC/TCOVER,NCHAN,INSTRU(20),WATER    ,NSPECV,NSPECA,NORGAN,
     1 NFRACT,NELEM,NOLIT ,NCHECK,IDAY, ATOT, ATOTO,IYRDAY,NREPET(20)
     2,NCOH(20),LISCOH(98),NCOHCU(20),NCOHOR,NFRELM,NFRAC1,NSPCOH,NDEBUG     D180
     3 ,FLOUT,MICROB, BIOMIN(98),MONTH
C-----------------------------------------------------------------------
C      COMMON BLOCK /METEOR/ CONTAINS THE VALUES OF EXOGENOUS VARIABLES
C      FOR THE CURRENT TIME UNIT.
C-----------------------------------------------------------------------
      COMMON/METEOR/WIRRIG,ERO,RUNSOL(6),RUNDEB(3,6), DARAIN,DAYRUN,
     1EVAP,WATTEM,DAPHOT, DAYRAD, DADUST(3,6) , EXOG(98), RAINCO(6),
```

```
     2COMPIN(6), DETIN(5,6), RUNON,FLOWIN,DRIFTV(20,6,6), DRIFTA(99,6),
     3DRIFTM(3,6), DRIFPO(98)
      EQUIVALENCE (EXO,WIRRIG)
C-----------------------------------------------------------------------
C     THE NAME LIST CONTAINS THE PARAMETERS THAT ARE TO BE
C     READ IN AT EXECUTION TIME.
C-----------------------------------------------------------------------
      NAMELIST/EPUT/DRIFTV,DRIFTA,DRIFTM,LIMEX,LIMEXO
    1 FORMAT (16I5)
    2 FORMAT (8F10.2)
      MONOLD = 0.
C-----------------------------------------------------------------------
C     EXOGENOUS AND METEROLOGICAL DATA IS READ.
C-----------------------------------------------------------------------

C..........DATA ON INFLOWING WATER FROM UPSTREAM.
      READ (5,2211) FLOWIN
 2211 FORMAT (F15.3)
      READ (5,2) (COMPIN(I), I=1,NFRELM)
      DO 94 I = 1, NOLIT
   94 READ (5,2) (DETIN(I,J), J = 1, NFRELM)

C..........PRECIPITATION DATA.
      IF (NRAIN.LE.0) GO TO 10
      READ (5,1) (MRAIN(I), I=1,NRAIN)
      READ (5,2) ( RAIN(I), I=1,NRAIN)
      READ (5,2) (RAINCO(K), K=1,NELEM)
      DO 7 IRAIN = 1, NRAIN
      IF (IDAY.LE.MRAIN(IRAIN)) GO TO 28
    7 CONTINUE
   10 TRAIN = NRAIN + 1

C..........SURFACE FLOW DATA.
   28 IF (NRUNON.LE.0) GO TO 29
      READ (5,1) (MRUNON(K), K=1,NRUNON)
      READ (5,2) (RUNON(I), I=1,NRUNON)
      READ (5,2) (ERODED(I), I=1,NRUNON)
      DO 13 I = 1, NRUNON
      READ (5,2) (RUNMIN(I,K), K= 1,NELEM)
      DO 13 J=1,NOLIT
   13 READ (5,2) (RUNORG(I,J,K),K=1,NFRELM)
      DO 14 IRUN = 1, NRUNON
      IF (IDAY.LE.MRUNON(IRUN)) GO TO 11
   14 CONTINUE
   29 TRUN = NRUNON + 1

C..........WATER REMOVAL DATA.
   11 IF (NIRRIG.LE.0) GO TO 24
      READ (5,1) (MIRRIG(K),K=1,NIRRIG)
      READ (5,2) (WATIRR(K),K=1,NIRRIG)
      DO 26 IRRIG=1,NIRRIG
      IF (IDAY.LE.MIRRIG(IRRIG)) GO TO 27
   26 CONTINUE
   24 TRRIG=NIRRIG + 1

C..........EVAPORATION, PHOTOPERIOD, RADIATION, WATER TEMPERATURE.
   27 READ (5,2) (EVAPOR(I), I=1,12)
```

```
      READ (5,2) (PHOTOP(I), I=1,12)
      READ (5,2) (RADIA (I), I=1,12)
      READ (5,2) (WTEMP(I),I=1,12)
C-----------------------------------------------------------------------
C     CHECK FOR DAILY EVENTS AND TRANSFER INFORMATION TO METEOR
C     COMMON BLOCK.
C-----------------------------------------------------------------------
      ENTRY EXOGE2
      DO 3 I = LIMEX, LIMEXO
    3 EXO(I) = 0.
   17 IF ((IRAIN.GT.NRAIN).OR.(IDAY.NE.MRAIN(IRAIN)))GO TO 20
      DARAIN = RAIN(IRAIN)
      IRAIN = IRAIN + 1
   20 IF((IRUN.GT.NRUNON).OR.(IDAY.NE.MRUNON(IRUN))) GO TO 19
      DAYRUN= RUNON(IRUN)
      ERO =ERODED(IRUN)
      DO 21 K = 1, NELEM
   21 RUNSOL(K) = RUNMIN(IRUN,K)
      DO 22 K = 1, NFRELM
      DO 22 J=1,NOLIT
   22 RUNDEB(J,K)=RUNORG(IRUN,J,K)
      IRUN = IRUN + 1
   19 IF ((IRRIG.GT.NIRRIG).OR.(IDAY.NE.MIRRIG(IRRIG))) GO TO 25
      WTRRIG=WATIRR(IRRIG)
      IRRIG=IRRIG+1
C-----------------------------------------------------------------------
C     TRANSFER NEW METEOROLOGICAL AVERAGES AT MONTHLY INTERVALS.
C-----------------------------------------------------------------------
   25 IF (MONTH.EQ.MONOLD) RETURN
      EVAP = EVAPOR(MONTH)
      DAYRAD = RADIA(MONTH)
      DAPHOT = PHOTOP(MONTH)
      WATTEM=WTEMP(MONTH)
      MONOLD = MONTH
      RETURN
      ENTRY EINPUT
      READ (5,EPUT)
      RETURN
      END
```

```
      SUBROUTINE REPORT
      COMMON /OTHER/PRECMM,TOTAL(5),CFRAC(5),SOURCE(6,3)
C-----------------------------------------------------------------------
C     THE COMMON BLOCK /NAMES/ CONTAINS THE NAMES REQUIRED FOR
C     TABULATED OUTPUT
C-----------------------------------------------------------------------
      COMMON /NAMES/COHNAM(21,4),RACNAM(3,4),VSPNAM(20,7),ASPNAM(20,7),
     1 ORGNAM(6,6),FRANAM(10,4),ALINAM(5,4)
C-----------------------------------------------------------------------
C     COMMON BLOCK /ACC/ CONTAINS ACCUMULATED CHANGES, WHICH MAY BE
C     NEGATIVE.  COMMON BLOCK /ACCINC/ CONTAINS THE INCREMENTS TO THE
C     ARRAYS IN /ACC/ FOR A SINGLE TIME UNIT.
C-----------------------------------------------------------------------
      COMMON /ACC/ AGAIN(3,5),EROD(7),H2O(7)                            0180
      COMMON /ACCINC/ AGAINQ(3,5),ERODQQ(3), H2OQQQ(3)                  0200
C-----------------------------------------------------------------------
C     COMMON BLOCK /SPEC/ CONTAINS SPECIFICATIONS AND OTHER INFORMATION
C     COMMON TO THE WHOLE SET OF PROGRAMS, BUT EXCLUDING STATE AND
C     EXOGENOUS VARIABLES.
C-----------------------------------------------------------------------
      COMMON/SPEC/TCOVER,NCHAN,INSTRU(20),    WATER,NSPECV,NSPECA,NORGAN,  0220
     1 NFRACT,NELEM,NOLIT  ,NCHECK,IDAY, ATCT, ATCTO,IYRDAY,NREPET(20)  0240
     2,NCOH(20),LISCOH(98),NCOHCU(20),NCOHOR,NFRELM,NFRAC1,NSPCOH,NDEBUG 0260
     3,FLOUT,MICROB  ,BIOMIN(98)                                        0280
C-----------------------------------------------------------------------
C     COMMON BLOCK /TOTALS/ CONTAINS SUMS OF THE STATE VARIABLES,
C     TOGETHER WITH CERTAIN OTHER VARIABLES REQUIRING INITIALIZATION
C     BUT NOT INCREMENTATION AT EACH TIME UNIT.
C-----------------------------------------------------------------------
      COMMON/TOTALS/CVEGV(6,6),CVEGO(20,6),CVEGVO(6),AVEGV(6 ),AVEGO(20  0300
     1),AVEGVO,ABIOMA,CBIOMA(6),ALITT,CLITT(6),CORGT(6)     ,ABIOSP(20), 0320
     2TOT(6),AORGT,POPSP(20),ANIM(20,6),CBACTT(6),ABACT(3),ABACTT,      0340
     3 AVEG(20,6),ABIOM(38),ALIT(5),AORG(5),AMIN                        0360
C-----------------------------------------------------------------------
C     COMMON BLOCK /STAT/ CONTAINS THE STATE VARIABLES, AND /CHANGE/
C     THEIR INCREMENTS OR DECREMENTS FOR THE CURRENT TIME UNIT.
C-----------------------------------------------------------------------
      COMMON/STAT/ CVEG(20,6,6),CORG(5,6),POP(98),CBIOM(98,6),AQUA(6),  0380
     1 CLIT(5,6),CBACT(3,6),            DUMMY(96)
      COMMON /CHANGE/ CVEGQQ(20,6,6),CORGQQ(5,6),POPQQQ(98),CBIOMQ(98,6)  042
     1,AQUAQQ(6), CLITQQ(5,6),CBACTQ(3,6)             ,DUMMQ(96)
C-----------------------------------------------------------------------
C     COMMON BLOCK /DIAGR/ CONTAINS INFORMATION REQUIRED FOR GRAPHS.
C-----------------------------------------------------------------------
      COMMON /DIAGR/FIG(8,70), EXPLA(5,8),  TITLE(20),  YTITLE(10),
     1             XDOT(80),         XMAX,XMIN,YMAX,YMIN,NOSYM,NODOT,NOCOL  0480
C-----------------------------------------------------------------------
C     COMMON BLOCK /METEOR/ CONTAINS THE VALUES OF EXOGENOUS VARIABLES
C     FOR THE CURRENT TIME UNIT.
C-----------------------------------------------------------------------
      COMMON/METEOR/WIRRIG,ERO,RUNSOL(6),RUNDEB(3,6),DARAIN,DAYRUN,     0500
     1EVAP,WATTEM,DAPHOT,DAYRAD,DADUST(3,6),EXOG(98),RAINCO(6),         0520
     2COMPIN(6),DETIN(5,6),RUNON,FLOWIN,DRIFTV(20,6,6),DRIFTA(98,6),    0540
     3DRIFTM(3,6),DRIFPO(98)                                            0560
C-----------------------------------------------------------------------
C     COMMON BLOCK /PARAM/ CONTAINS THE VALUES OF PARAMETERS USED BY THE
```

```
C       PROCESS SUBROUTINES.
C-------------------------------------------------------------------------------
        COMMON /PARAM/ P(10200)
        DIMENSION   AGAINT(3)
        DATA BLANK/' '/, TOTNAM/'TOTA'/, TOTNAI/'L   '/                 0700
        DATA CARBON/'CARB'/,CARBO1/'ON  '/                             0740
        DATA BICARB/'BICA'/, BICAR1/'RBON'/, BICAR2/'ATE '/
        DATA SOURCE/'TO O','R FR','OM A','TMOS','PHER','E   ','BY R',  0760
       1 'UN-O','FF O','F RU','N-ON',' ','TO O','R FR','OM S','UPSO',  0780
       2 'IL  ',' '/                                                   0800
        NFREL3=3*NFRELM
        NELEM3=3*NELEM
  2380 IF (NOSPEC.LE.0) GO TO 2420                                     9100
        TIME =  EXTIME(9)                                              9120
        WRITE (6,2400) TIME                                            9140
  2400 FORMAT ('+', 103X, F10.3, ' SECONDS ELAPSED')                   9160
C-------------------------------------------------------------------------------
C       THE STATE VARIABLES ARE PRINTED.
C-------------------------------------------------------------------------------

C..........PLANT CONSTITUENTS
  2420 IF (NSPECV.LE.0) GO TO 2760                                       9180
        WRITE (6,2440)                                                   9200
  2440 FORMAT('0CONSTITUENTS OF PRIMARY PRODUCERS, G.(OR KCAL.) PER SQ.M. 9220
       1')                                                                9240
        IF(NFRACT.GE.2)WRITE (6,2460)(BLANK,I=1,NELEM3),(CFRAC(I),I=1,5)  9260
  2460 FORMAT (30X, 32A4)                                                9280
        WRITE (6,2800) ((FRANAM(I,J), J = 1,3), I = 1,NFRELM)            9300
        IF(NFRACT.GT.1) WRITE(6,2480)(BLANK,J=1,NFREL3),TOTNAM,TOTNAI    9320
  2480 FORMAT ('+', 30X, 32A4)                                           9340
        DO 2620 I = 1, NSPECV                                            9360
        WRITE (6,380) (VSPNAM(I,J), J = 1, 7)                            9380
   380 FORMAT(1X,30A4)
        IF (NORGAN.GT.1) GO TO 2540                                      9400
        IF(NFRACT.GT.0) GO TO 2500                                       9420
        WRITE (6,2520) (CVEG(I,1,K), K=1,NFRELM)                         9440
        GO TO 2620                                                       9460
  2500 WRITE (6,2520) (CVEG(I,1,K), K=1,NFRELM), AVEG(I,1)               9480
  2520 FORMAT ('+', 26X, 8F12.5)                                         9500
        GO TO 2620                                                       9520
  2540 DO 2560 J = 1, NORGAN                                             9540
  2560 WRITE (6,2580) (ORGNAM(J,K), K = 1,6), (CVEG (I,J,K),K=1,NFRELM), 9560
       1AVEG(I,J)                                                        9580
  2580 FORMAT(3X, 6A4,    8F12.5)                                        9600
        IF(NFRACT.GT.0)GO TO 2600                                        9620
        WRITE (6,2640) (CVEGO(I,K), K=1,NFRELM)                          9640
        GO TO 2620                                                       9660
  2600 WRITE (6,2640) (CVEGO(I,K), K=1,NFRELM), AVEGO(I)                 9680
  2620 CONTINUE                                                          9700
  2640 FORMAT (12X,'TOTAL', 10X,8F12.5)                                  9720
        WRITE (6,2660)                                                   9740
  2660 FORMAT ('0ALL SPECIES')                                           9760
        IF(NORGAN.LE.1) GO TO 2720                                       9780
        DO 2700 J=1,NORGAN                                               9800
        IF(NFRACT.GT.0)GO TO 2680                                        9820
        WRITE(6,2580)(ORGNAM(J,K),K=1,6),(CVEGV(J,K),K=1,NFRELM)         9840
        GO TO 2700                                                       9860
```

```
 2680 WRITE(6,2580)(ORGNAM(J,K),K=1,6),(CVEGV(J,K),K=1,NFRELM),AVEGV(J)
 2700 CONTINUE
 2720 IF(NFRACT.GT.0)GO TO 2740
      WRITE (6,2640) (CVEGVO(K), K = 1,NFRELM)
      GO TO 2760
 2740 WRITE (6,2640) (CVEGVO(K), K = 1,NFRELM),AVEGVO

C..........ANIMAL CONSTITUENTS BY SPECIES AND COHORT.
 2760 IF (NSPECA.LE.0) GO TO 3200
      WRITE (6,2780)
 2780 FORMAT ('0CONSTITUENTS OF ANIMAL BIOMASS, G.(OR KCAL.) PER SQ.M.')
      IF(NFRACT.GE.2)WRITE (6,2460)(BLANK,I=1,NELEM7),(CFRAC(I),I=1,5)
      WRITE (6,2800) ((FRANAM(I,J), J = 1,3), I = 1,NFRELM)
      IF(NFRACT.GT.1) WRITE(6,2430)(BLANK,J=1,NFREL3),TOTNAM,TOTNAI
 2800 FORMAT (30X, 24A4)
      DO 3000 I = 1, NSPECA
      K1=1
      IF(I.GT.1) K1=NCOHCU(I-1)+1
      K2=NCOHCU(I)
      WRITE(6,2820) (ASPNAM(I,J), J=1,7)
 2820 FORMAT (' ', 7A4)
      IF(K2.GT.K1) GO TO 2880
      IF(NFRACT.GT.0)GO TO 2840
      WRITE (6,2860) (CBIOM(K1,J), J=1,NFRELM)
      GO TO 3000
 2840 WRITE (6,2860) (CBIOM(K1,J), J=1,NFRELM), ABIOM(K)
 2860 FORMAT ('+',26X, 8F12.5)
      GO TO 3000
 2880 DO 2920 K = K1, K2
      L=LISCOH(K)
      IF(NFRACT.GT.0)GO TO 2900
      WRITE(6,2940)(COHNAM(L,J),J=1,4),(CBIOM(K,J),J=1,NFRELM)
      GO TO 2920
 2900 WRITE(6,2940)(COHNAM(L,J),J=1,4),(CBIOM(K,J),J=1,NFRELM),ABIOM(K)
 2920 CONTINUE
 2940 FORMAT (5X, 4A4, 6X, 8F12.5)
      IF(NFRACT.GT.0)GO TO 2960
      WRITE (6,3080) (ANIM(I,K),K=1,NFRELM)
      GO TO 2980
 2960 WRITE (6,3080) (ANIM(I,K),K=1,NFRELM), ABIOSP(I)
 2980 WRITE (6,3020)
 3000 CONTINUE
 3020 FORMAT (1H )
      IF(NFRACT.GT.0)GO TO 3040
      WRITE (6,3060) (CBIOMA(K), K = 1, NFRELM)
      GO TO 3100
 3040 WRITE (6,3060) (CBIOMA(K), K = 1, NFRELM), ABIOMA
 3060 FORMAT ('0TOTAL, ALL SPECIES', 8X, 8F12.2)
 3080 FORMAT (10X, 'TOTAL',12X,8F12.5)

C..........ANIMAL NUMBERS BY SPECIES AND COHORT.
 3100 WRITE (6,3120)
 3120 FORMAT ('0ANIMAL POPULATIONS, PER SQ. M.')
      DO 3180 I = 1, NSPECA
      K1 = 1
      IF (I.GT.1) K1 = NCOHCU(I-1) + 1
      K2 = NCOHCU(I)
```

```
      WRITE(6,2820) (ASPNAM(I,J), J=1,7)
      IF(K2.GT.K1) GO TO3140
      WRITE (6,3130) POP(K1)
 3130 FORMAT('+', 26X,8F13.4)
      GO TO 3180
 3140 DO 3160 K = K1, K2
      L=LISCOH(K)
 3160 WRITE (6,3161) (COHNAM(L,J),J=1,4),POP(K)
 3161 FORMAT (5X,4A4,6X,8F13.4)
      WRITE (6,3070) POPSP(I)
 3070 FORMAT (10X,'TOTAL', 12X,8F13.4)
      WRITE (6,3020)
 3180 CONTINUE

C..........CONSTITUENTS OF HETEROTROPHIC MICRO-ORGANISMS.
 3200 WRITE (6,3220)
 3220 FORMAT ('0CONSTITUENTS OF HETEROTROPHIC MICRO-ORGANISMS, G. (OR KC
     1AL.) PER SQ. M.')
      IF (NFRACT.GE.2) WRITE (6,2460) (BLANK,I=1,NELEM3),(CFRAC(I),
     1I=1,5)
      WRITE (6,3240) ((FRANAM(I,J),J=1,3),I=1,NFRELM)
      IF (NFRACT.GE.2) WRITE (6,2480) (BLANK,J=1,NFREL3),TOTNAM,TOTNAI
 3240 FORMAT ('   MICROBIAL TYPE',12X,24A4)
      DO 3280 I=1,MICROB
      IF (NFRACT.GE.2) GO TO 3260
      WRITE (6,2940) (BACNAM(I,J),J=1,4),(CBACT(I,J),J=1,NFRELM)
      GO TO 3280
 3260 WRITE (6,2940) (BACNAM(I,J),J=1,4),(CBACT(I,J),J=1,NFRELM),ABACT(I
     1)
 3280 CONTINUE

C..........SUSPENDED DETRITUS CONSTITUENTS.
      IF (NFRACT.GE.2) GO TO 3300
      WRITE (6,2640) (CBACTT(K),K=1,NFRELM)
      GO TO 3320
 3300 WRITE (6,2640) (CBACTT(K),K=1,NFRELM),ABACTT
 3320 WRITE (6,3340)
 3340 FORMAT('0SUSPENDED DETRITUS CONSTITUENTS, G.(OR KCAL.) PER SQ.M.')
      IF(NFRACT.GE.2)WRITE (6,2460)(BLANK,I=1,NELEM3),(CFRAC(I),I=1,5)
      WRITE (6,3360) ((FRANAM(I,J), J = 1,3), I = 1,NFRELM)
      IF(NFRACT.GT.1) WRITE(6,2480)(BLANK,J=1,NFREL3),TOTNAM,TOTNAI
 3360 FORMAT (3X,'DETRITUS TYPE', 13X, 24A4)
      DO3400 I = 1, NOLIT
      IF(NFRACT.GT.0)GO TO 3380
      WRITE (6,2940) (ALINAM(I,J), J=1,4),(CLIT(I,J),J=1,NFRELM)
      GO TO 3400
 3380 WRITE (6,2940) (ALINAM(I,J), J=1,4),(CLIT(I,J),J=1,NFRELM),ALIT(I)


C..........BIOLOGICALLY ACTIVE SEDIMENTS.
 3400 CONTINUE
 3420 FORMAT (1X, 4A4, 10X, 8F12.5)
      IF(NFRACT.GT.0)GO TO 3440
      WRITE (6,2640) (CLITT(K),K=1,NFRELM)
      GO TO 3460
 3440 WRITE (6,2640) (CLITT(K),K=1,NFRELM),ALITT
 3460 WRITE (6,3480)
```

```
 3480 FORMAT ('0BIOLOGICALLY ACTIVE SEDIMENTS, G.(OR KCAL.) PER SQ.M.')
      IF(NFRACT.GE.2)WRITE (6,2460)(BLANK,I=1,NELEM3),(CFRAC(I),I=1,5)
      WRITE (6,3360) ((FRANAM(I,J), J = 1,3), I = 1,NFRELM)
      IF(NFRACT.GT.1) WRITE(6,2480)(BLANK,J=1,NFREL3),TOTNAM,TOTNAI
 3500 FORMAT (34X, 24A4)
      DO 3540 I=1,NOLIT
      IF(NFRACT.GT.1)GO TO 3520
      WRITE(6,2940)(ALINAM(I,J),J=1,4),(CORG(I,J),J=1,NFRELM)
      GO TO 3540
 3520 WRITE(6,2940)(ALINAM(I,J),J=1,4),(CORG(I,J),J=1,NFRELM),AORGT
 3540 CONTINUE
      IF(NFRACT.GT.1)GO TO 3560
      WRITE(6,2640)(CORGT(K),K=1,NFRELM)
      GO TO 3580
 3560 WRITE(6,2640)(CORGT(K),K=1,NFRELM), AORGT
 3580 IF(NFRACT.GT.1)GO TO 3600
      WRITE (6,3620) (TOT(K), K = 1,NFRELM)
      GO TO 3640
 3600 WRITE (6,3620) (TOT(K), K = 1,NFRELM),ATOT
 3620 FORMAT ('0TOTAL,SEDIMENTS + DETRITUS',8F12.5)

C---------DISSOLVED CONSTITUENTS IN THE WATER.
 3640 WRITE (6,3650) (BLANK, I = 1, NELEM3), BICARB,BICAR1,BICAR2
 3650 FORMAT (27X,24A4)
      WRITE (6,3650) (BLANK, I = 1, NELEM3), CARBON, CARB01
      WRITE (6,3660)(AQUA(I), I = 1, NFRAC1)
 3660 FORMAT ('0IN WATER', 18X, 8F12.5)

C..........TOTAL CONSTITUENTS IN THE ECOSYSTEM.
      IF (NFRACT.GT.1) GO TO 3700
      WRITE (6,3680)(TOTAL(K), K = 1,NFRELM)
 3680 FORMAT (    '0TOTAL IN ECOSYSTEM', 8X, 8F12.5)
      GO TO 3720
 3700 WRITE (6,3680)(TOTAL(K), K = 1,NFRELM), ATOTO

C..........NET GAIN OR LOSS TO THE ECOSYSTEM.
 3720 IF (NCHECK.EQ.0) GO TO 1780
      WRITE (6,3740)
 3740 FORMAT (///'0ACCUMULATED NET GAIN OR')
      WRITE (6,3760)     ((FRANAM(K,J),J=1,3),K=1,NELEM)
 3760 FORMAT (' LOSS TO ECOSYSTEM',19X,'WATER   INERT PARTICLES ', 18A4)
      IF (NFRACT.LE.0) GO TO 3800
      WRITE (6,3780) (BLANK, K=1,NELEM3), CARBON, CARB01
 3780 FORMAT ('+', 61X, 18A4)
 3800 J = NELEM
      IF (NFRACT.GT.0) J = J + 1
      DO 3820 I = 1,NCHAN
 3820 WRITE(6,3840)(SOURCE(L,I),L=1,6),H2O(I),EROD(I),(AGAIN(I,L),L=1,J)
 3840 FORMAT (4X, 6A4, F16.5, 7F12.5)
      H2OTOT = 0.
      EROTOT = 0.
      DO 3860 I = 1, NELEM
 3860 AGAINT(I) = 0.
      DO 3880 I = 1,NCHAN
      H2OTOT = H2OTOT + H2O(I)
      EROTOT = EROTOT + EROD(I)
      DO 3880 K = 1, J
```

```
 3880 AGAINT(K) = ACAINT(K) + AGAIN (I,K)                                2880
      WRITE (6,3900) H2OTOT, EROTOT, (AGAINT(K), K=1,NELEM)             2900
 3900 FORMAT ('0   TOTAL', 13X, F16.5, 7F12.5)                          2920
      IF (NCHECK.EQ.0) GO TO 1780                                       2940
      WRITE (6,3920) PRECMM,PRECMM                                      2980
 3920 FORMAT ('0ACCUMULATED PRECIPITATION =', F7.1,' MM. - THAT IS, ',  3000
     1 3PF14.1,' G. PER SQ.M.')                                         3020
 1780 RETURN
      END
```

```
          SUBROUTINE GRAF
C         WHEN CALLED AS GRAF , THIS SUBROUTINE RECEIVES IN EACH ROW OF THE
C         ARRAY FIGS THE 70 VALUES OF Y REQUIRED FOR SUCCESSIVE VALUES OF X,
C         EQUALLY SPACED BETWEEN XMIN AND XMA.   THE MINIMUM AND MAXIMUM
C         VALUES OF Y ARE TRANSFERRED AS YMIN, YMAX.    THE NUMBER OF ROWS
C         OF FIGS USED (I.E. THE NUMBER OF CURVES REQUIRED ON THE GRAPH) IS
C         TRANSFERRED AS NOSYM.    THE TITLE FOR THE GRAPH IS TRANSFERRED AS
C         TITLE, THAT FOR THE Y AXIS AS YTITLE.
C         THE MEANINGS OF THE DIFFERENT GRAPHS ARE TRANSFERRED (IN ORDER) IN
C         THE SUCCESSIVE COLUMNS OF EXPLAN.   ENTRY HIST PRODUCES A HISTO-
C         -GRAM WITH NOCOL COLUMNS EQUALLY SPACED BETWEEN XMIN AND XMA, THE Y VALUES
C         FOR EACH COLUMN BEING TRANSFERRED IN XDOT.    OTHER VARIABLES AND ARRAY
C         NAMES HAVE THE SAME MEANING THROUGHOUT.
          COMMON/DIAGR/FIGS (8,70),EXPLAN(5, 8),TITLE(20),YTITLE(10),
         1           XDOT(80),          XMAX,XMIN,YMAX,YMIN,NOSYM,NODOT,NOCOL    D48D
         2 ,IYP,INITYR
          DIMENSION FMT1(7),SYMBOL(8),GRAPH(51,71),ANUM(10),YAXIS(6),
         1 XLINE(18),IDAYS(8),IYEARS(8)
          DATA XLINE/'+---','----','--+-','2*'----','+---','-----','--+-',
         1 2*'----','+---','----','--+-',2*'----','+---','-----','--+ '/
          DATA ANUM /'0','1','2','3','4','5','6','7','8','9'/
          DATA FMT1/'(1H+','','',' ',' ','PF18','.4,8','CA1)'/
          DATA BLANK /' '/ STAR /'*'/  SMALL/3.E-8/
          DATA APOS/'I' /  PLUS/'+'/ HYPHEN /'-'/
          DATA SYMBOL/'A','B','C','D','E','F','G','H'/
          INDEX = 1
          GO TO 655
          ENTRY HIST
          NOSYM = 0
          INDEX = 3
          INTERV = 71/NOCOL
          ISTART = MAX0(1,INTERV/2)
          IEND = ISTART + INTERV*(NOCOL-1)
      655 FMT1(4) = BLANK
          FMT1(3) = BLANK
          IF (XMAX.GT.XMIN) GO TO 101
          WRITE (6,10)
       10 FORMAT ('1NO GRAPH POSSIBLE BECAUSE NO TIME EXPIRED')
          RETURN
      101 IF (YMAX.NE.YMIN) GO TO 102
          IF (YMAX.NE.0.) GO TO 1101
          YMAX = SMALL
          YMIN = - SMALL
          GO TO 102
     1101 YMAX = YMAX * 1.001
          YMIN = YMIN * .999
      102 DO 938 I = 1, 51
          DO 937 J = 1, 71
      937 GRAPH(I,J) = BLANK
      938 CONTINUE
          B = AMAX1(ABS(YMIN), ABS(YMAX))
          T = 0
          A = XMAX - XMIN
          IF ((A.GT.0.).AND.(B.GT.0.).AND.(YMAX.GT.YMIN)) GO TO 383
          WRITE (6,5376) XMAX, XMIN, YMAX, YMIN
     5376 FORMAT ('0ERROR IN LIMITS:', 4E20.6)
```

```
      RETURN
  383 IF (B.GE.1.) GO TO 381
      B = B * 10.
      I = I - 1
      GO TO 383
  381 IF (B.LT.10.) GO TO 382
      B = B * .1
      I = I + 1
      GO TO 381
  382 I102 = I
      J = IABS(I)
      IF (J.LE.9) GO TO 2291
      WRITE (6,301) (TITLE(K), K=1,20)
      WRITE (6,2292)
 2292 FORMAT ('0FACTOR FOR Y AXIS .GT. 10**9 OR .LT. 10**-9')
      RETURN
 2291 IF (I.LE.0) GO TO 393
      FMT1(3) = HYPHEN
  393 FMT1(4)      = ANUM(J+1)
      DO 1 I = 1, 51
    1 GRAPH(I,1) = APOS
      DO 3 I = 1,51,10
    3 GRAPH(I,1) = PLUS
      YUNIT = 50./(YMAX-YMIN)
      XUNIT= 70./(XMAX - XMIN)
      GO TO (901,902 , 903              ), INDEX
  902 WRITE (6,911)
  911 FORMAT ('0DOT DIAGRAM FACILITY NOT AVAILABLE')
      RETURN
  903 Y = YMIN
      YUN = 1./YUNIT
      DO 921 I = 1, 51
      DO 811 K = 2, NOCOL
      K1 = ISTART + (K - 1) * INTERV
      IF((K1.LE.0).OR.(K1.GT.71))  WRITE (6,7524) I, K, K1
 7524 FORMAT (' I =', I5, ' K =', I5,'K1 =', I5)
      IF (XDOT(K).GE.Y) GRAPH(I,K1) = STAR
  811 CONTINUE
      Y = Y + YUN
  921 CONTINUE
      GO TO 912
  901 DO 34 I = 1, NOSYM
      DO 5 J = 1, 70
      K = (FIGS(I,J) - YMIN)*YUNIT+ .1
      IF (K.GT.50) GO TO 5
      GRAPH(K+1,J+1)= SYMBOL(I)
    5 CONTINUE
   34 CONTINUE
  912 XUNIT = (XMAX - XMIN)/7.
      IDAYS(1) = XMIN
      IYEARS(1) = INITYR
      DO 16 I = 2,8
   16 IDAYS(I) = XMIN + XUNIT * FLOAT(I-1)
      DO 6 I = 2, 8
      IYEARS(I) = IYEARS(I-1)
   11 NYRDAY = 365
      IF (MOD(IYEARS(I),4).EQ.0) NYRDAY = 366
```

```
      IF (IDAYS(I).LE.NYRDAY) GO TO 6
      IYEARS(I) = IYEARS(I) + 1
      DO 15 J = I,8
   15 IDAYS(J) = IDAYS(J) - NYRDAY
      GO TO 11
    6 CONTINUE
      YUNIT = (YMAX - YMIN )/5.
      YAXIS(1) = YMAX
      DO 7 J = 2, 6
    7 YAXIS(J) = YAXIS(J-1) - YUNIT
      WRITE (6,301) (TITLE(I), I = 1, 20)
  301 FORMAT (1H1, 20A4)
      IF (NOSYM.GT.1)WRITE(6,1011)SYMBOL(1),(EXPLAN(I,1),I=1,5)
      WRITE (6,1021)
 1021 FORMAT (1H )
      IF (NOSYM.GT.1)WRITE(6,1011)SYMBOL(2),(EXPLAN(I,2),I=1,5)
 1011 FORMAT (1H+, 95X, 6A4)
      WRITE (6,303) I102, (YTITLE(I), I = 1,10)
  303 FORMAT ('  Y AXIS (*10**',I2,') IS ',10A4)
 1012 J = 1
       I3=2
      DO 21 I1= 1, 51
      I3 = I3 + 1
      I = 52 - I1
      WRITE (6,9) (GRAPH(I,K), K = 1, 71)
      IF (I-I/10*10.NE.1) GO TO 121
      WRITE (6,FMT1) YAXIS(J)
      J = J + 1
  121 IF(I3.LE.NOSYM)WRITE(6,1011)SYMBOL(I3),(EXPLAN(K,I3),K=1,5)
   21 CONTINUE
      WRITE (6,89) (XLINE(I), I = 1,13)
   89 FORMAT (20X, 13A4)
      WRITE (6,12) (IDAYS(I), I = 1,8)
   12  FORMAT ('   TIME - DAY   ',I5, 7(6X,I4))
      WRITE (6,13)  IYR
   13 FORMAT (9X,'YEAR    ',I5)
    9 FORMAT (20X, 80A1)
      RETURN
      END
```

```
      SUBROUTINE SENSIT(IRUN,NRA,NRUN)
      COMMON/PARAM/P(17377)
      DIMENSION NDIF(50),PDIF(50,10),IPAR(50),IPA(10),IPB(10)
      DIMENSION ITYP(20),INUM(20),IDAYT(20),IDPAR(300),IDPAQ(300),
     1VPAR(300),VPAQ(300) ,PARX(30) ,IQA(300),IQB(300)
      DIMENSION IY(50)
      COMMON/STAT/ STATE(1586)
      COMMON/TOTALS/ SUMS(615)
      COMMON/ACC/STNG(21)
      COMMON /RESP/ ITYP,INUM,IDAYT,PARX,NRESP,NPAR,NRU
      NRU=NRUN
      IF(IRUN.GT.1)GO TO 10
      WRITE(8)STATE,P
C READ PARAMETERS TO BE TESTED AND MODIFIED VALUES
      READ (5,1) NPAR,NINTER,NRESP
    1 FORMAT(16I5)
      NRA=1
      DO 3 I=1,NPAR
      READ(5,2) ITY   ,IP,NDI,(PDIF(I,J),J=1,NDI)
      IPAR(I)=IP
      NDIF(I)=NDI
      GOTO(51,52),ITY
   51 PARX(I)=P(IP)
      GOTO53
   52 PARX(I)=STATE(IP)
   53 CONTINUE
      DO 33 J=1,NDI
      NRA=NRA+1
      IDPAR(NRA)=IP
      VPAR(NRA)=PDIF(I,J)
      IQA(NRA)=I
      IY(I)=ITY
   33 CONTINUE
    3 CONTINUE
    2 FORMAT(3I5,8F8)
      NRI=0
      NRU =NRA
C READ PARAMETER PAIRS FOR INTERACTION TEST
      IF(NINTER.EQ.0)GOTO6
      DO 4 I=1,NINTER
      READ (5,1) IPA(I),IPB(I)
      IA=IPA(I)
      IB=IPB(I)
      NNA=NDIF(IA)
      NNB=NDIF(IB)
      DO 34 J=1,NNA
      DO 34 K=1,NNB
      NRI=NRI+1
      NRU =NRU +1
      IDPAR(NRU)=IPAR(IA)
      VPAR(NRU)=PDIF(IA,J)
      IDPAQ(NRU)=IPAR(IB)
      VPAQ(NRU)=PDIF(IB,K)
      IQA(NRU)=IA
      IQB(NRU)=IB
   34 CONTINUE
```

```
    4 CONTINUE
C READ IDENTIFICATION OF RESPONSE VARIABLES
    6 CONTINUE
      NRUN=NRU
      DO 5 I=1,NRESP
      READ (5,1) ITYP(I),INUM(I),IDAYT(I)
      INU=INUM(I)
      ISV=1
      IF(ITYP(I).EQ.5)CALL DERIVD(INU,ISV)
    5 CONTINUE
      GO TO 99
   10 REWIND 8
      READ(8) STATE,P
      DO82I=1,NPAR
      IP=IPAR(I)
      PARX(I)=P(IP)
      IF(IY(I).EQ.1)GOTO82
      PARX(I)=STATE(IP)
   82 CONTINUE
      IF(IRUN.GT.NRA)GOTO20
      IDA=IDPAR(IRUN)
      IQ=IQA(IRUN)
      P    (IDA)=VPAR(IRUN)
      PARX(IQ)=P    (IDA)
      IF(IY(IQ).EQ.1)GOTO20
      STATE(IDA)=VPAR(IRUN)
      PARX(IQ)=STATE(IDA)
      GO TO 99
   20 CONTINUE
      IQ=IQA(IRUN)
      IDA=IDPAR(IRUN)
      IDB=IDPAQ(IRUN)
      IQQ=IQB(IRUN)
      P(IDA)=VPAR(IRUN)
      PARX(IQ)=P(IDA)
      IF(IY(IQ).EQ.1)GOTO55
      STATE(IDA)=VPAR(IRUN)
      PARX(IQ)=STATE(IDA)
   55 P(IDB)=VPAQ(IRUN)
      PARX(IQQ)=P(IDB)
      IF(IY(IQQ).EQ.1)GOTO99
      STATE(IDB)=VPAQ(IRUN)
      PARX(IQQ)=STATE(IDB)
   99 CONTINUE
      RETURN
      END
```

SENOUT

PROGRAM LISTING

```
      SUBROUTINE SENOUT(ISW,IDAY,IRUN)
      COMMON/PARAM/P(17377)
      COMMON/NEWVAR/VNEW(10)
      DIMENSION ITYP(20),INUM(20),IDAYT(20)
      DIMENSION PARX(30),R(20)
      COMMON/STAT/ STATE(1036)
      COMMON/TOTALS/ SUMS(615)
      COMMON/ACC/STNG(21)
      COMMON/RESP/ITYP,INUM,IDAYT,PARX,NRESP,NPAR,NRUN
      IF(ISW.GT.3)ISW=1
      GOTO(1,2,3,1,1),ISW
    1 DO 4 I=1,NRESP
      IF(IDAY.NE.IDAYT(I))GO TO 4
      INA=INUM(I)
      IT=ITYP(I)
      ISV=2
      IF(IT.EQ.5)CALL DERIVD(INA,ISV)
      GOTO(11,12,13,14,15),IT
   11 R(I)=STATE(INA)
      GO TO 20
   12 R(I)=SUMS(INA)
      GO TO 20
   13 CONTINUE
      GO TO 20
   14 R(I)=STNG(INA)
      GOTO20
   15 R(I)=VNEW(INA)
   20 CONTINUE
    4 CONTINUE
      GO TO 99
    2 WRITE(8) IRUN,PARX,R
      PRINT110,IRUN
  110 FORMAT(1H ,I5)
      GO TO 99
    3 REWIND 8
      WRITE(6,101) (I,I=1,NPAR),(I,I=1,NRESP)
  101 FORMAT(1H1,15HSUMMARY OF RUNS/1H0,5X,10(6X,I6))
      DO 30 K=1,NRUN
      READ(8) IRUN,PARX,R
      WRITE(6,105)IRUN,(PARX(I),I=1,NPAR),(R(I),I=1,NRESP)
  105 FORMAT(1H ,I5,10F12.3)
   30 CONTINUE
   99 CONTINUE
      RETURN
      END
```

@PRT   WATER9.DERIVD

```
      SUBROUTINE DERIVO(INA,ISV)
      COMMON/STAT/ STATE(1586)
      COMMON/TOTALS/ SUMS(615)
      COMMON/ACC/STNG(21)
      COMMON/NEWVAR/VNEW(10)
      DIMENSION ITYP(10),NVAR(10),IVAR(10,10),WVAR(10,10)
      IF(ISV.EQ.2)GOTO10
      READ101,ITYP(INA),NVAR(INA)
      NV=NVAR(INA)
      READ101,(IVAR(INA,J),N=1,NV)
      READ102,(WVAR(INA,J),J=1,NV)
  101 FORMAT(16I5)
  102 FORMAT(8F10.4)
      RETURN
   10 CONTINUE
      VNEW(INA)=0.
      NV=NVAR(INA)
      DO16J=1,NV
      IV=IVAR(INA,J)
      IT=ITYP(INA)
      GOTO(11,12,13,14),IT
   11 Q=STATE(IV)
      GOTO15
   12 Q=SUMS(IV)
      GOTO15
   13 CONTINUE
      GOTO15
   14 Q=STNG(IV)
   15 VNEW(INA)=VNEW(INA)+Q*WVAR(INA,J)
   16 CONTINUE
      RETURN
      END
```

PROCESS SUBROUTINES

SUBROUTINE MEDIUM

David W. Goodall

2.1.3.1.2.-59

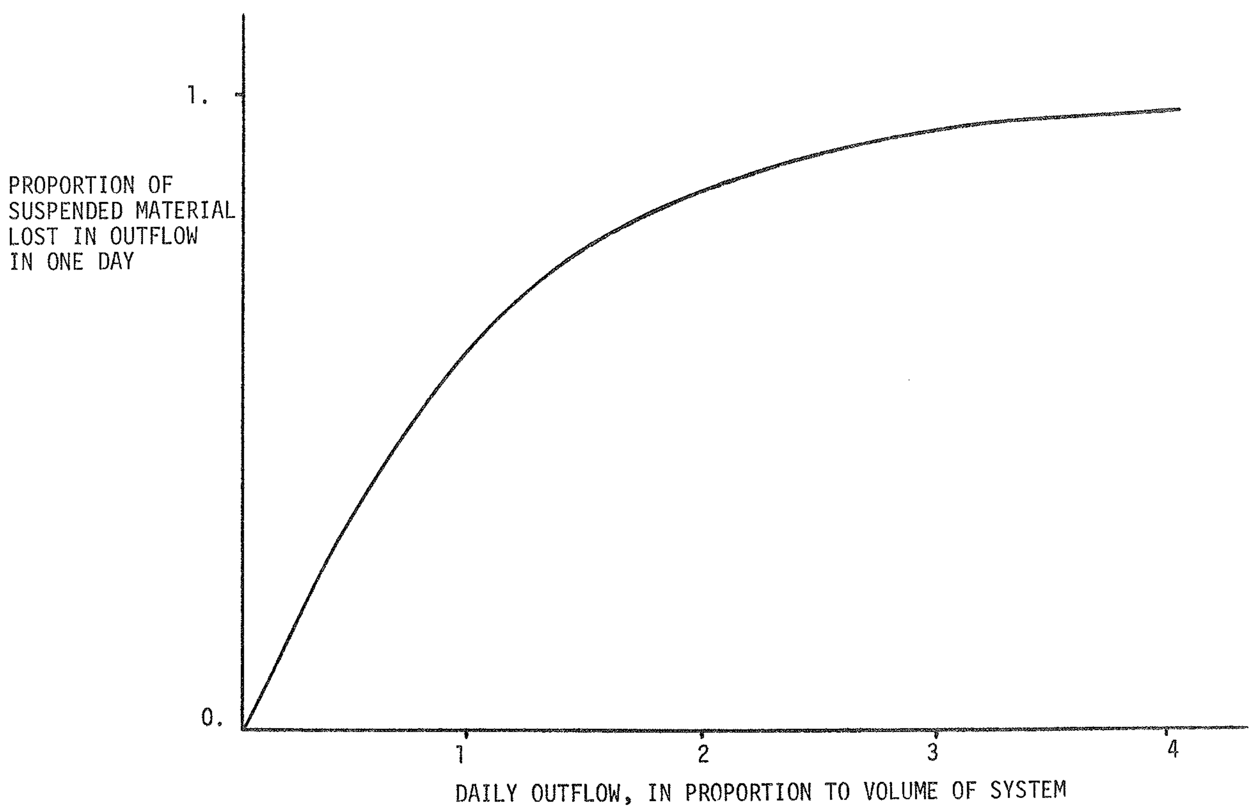
This subroutine, to be used in conjunction with the other modelling programs developed by the Desert Biome for aquatic ecosystems, models abiotic processes in the system -- water flow and the associated movement of sediment and detritus, and sedimentation of suspended material.

VERBAL DESCRIPTION

For water movement, the rates of inflow from different sources and of evaporation are given. The water level is assumed constant, so the outflow is simply a balancing term. The outflow carries with it any material which is in suspension, the proportion depending exponentially on the ratio of outflow water to the volume of the system (Fig. 1).

Input of sediment and detritus from above, or by overland flow, are incorporated into the system by this subroutine.

Suspended detritus sediments to the bottom at a rate specific to the detritus type (inorganic or organic, fine or coarse). The fine organic detritus is supplemented by coagulation of any organic carbon compounds in solution. All these processes take place at constant proportional rates.



Mathematical Description

Inflow and outflow

The water balance of the system is expressed in terms of the inflow at a rate of V_1, V_2 in the form of precipitation, V_3 as overland flow, and V_4 as evaporation, all expressed in grams per unit area. The water level is assumed constant, so the outflow is

$$_{O}\dot{X}_{012} = V_1 + V_2 + V_3 - V_4 \quad (1)$$

Substances in solution are carried in and out by the water, so

$$_{I}\dot{X}_{24_c} = V_1 V_{5_c} + V_2 V_{6_c} + V_3 V_{7_c} - {_{O}\dot{X}_{012}} X_{21_c} \quad (2)$$

Where V_{5_c}, V_{6_c} and V_{7_c} are the content of the c'th chemical constituent in the inflow water, precipitation and overland flow respectively. Water inflow or overland flow may also carry a load of detritus, expressed respectively as $V_{8_{dc}}$ and $V_{9_{dc}}$; there may also be detritus or dust falling from the atmosphere, the quantities of the constituents added per time unit in this way being $V_{10_{dc}}$. Consequently,

$$_{I}\dot{X}_{21_{dc}} = V_1 V_{8_{dc}} + V_3 V_{9_{dc}} + V_{10_{dc}} - {_{O}\dot{X}_{012}} X_{21_{dc}} \quad (3)$$

The resulting net loss or gain to the system of the c'th constituent from these processes is

$$\sum_r {_{I}\dot{X}_{0_{rc}}} = - {_{I}\dot{X}_{24_c}} - \sum_d {_{I}\dot{X}_{21_{dc}}} \quad (4)$$

where r is the route of exchange, most of these processes being by surface flow, through evaporation, precipitation and dust fall are exchanges with the atmosphere.

Sedimentation

Suspended detritus sediments to the bottom at a rate P_{1_d} specific to the detritus type. Thus:

$$_{S}\dot{X}_{21_{dc}} = P_{1_d} X_{21_{dc}} \quad (5)$$

$$_{S}\dot{X}_{22_{dc}} = - \dot{X}_{21_{dc}} \quad (6)$$

Coagulation

Organic matter insolution (or in colloidal form) may also coagulate at a rate P_2, so that

$$ {}_C\dot{X}_{24_1} = P_2 X_{24_1} \tag{7} $$

$$ {}_C\dot{X}_{21_{11}} = - {}_C\dot{X}_{24_1} \tag{8} $$

$$ {}_C\dot{X}_{24_5} = P_2 X_{24_5} \tag{9} $$

$$ {}_C\dot{X}_{21_{15}} = - {}_C\dot{X}_{24_5} \tag{10} $$

It is assumed that this transfer includes organic carbon and chemical energy, but that involvement of the other chemical components may be ignored.

Array Limitations

The only array peculiar to this subroutine is FALL, the dimension of which limits the number of types of suspended material that can be handled (NOLIT in the FORTRAN program).

Symbol Table

Symbol	FORTRAN	Equation
P_{1d}	FALL(D)	5
P_2	COAGUL	7, 9
V_1	FLOWIN	1, 2
V_2	DARAIN	1, 2
V_3	FUNON	1, 2
V_4	FLOUT	1, 2
V_{5c}	COMPIN(C)	2
V_{6c}	RAINCO(C)	2
V_{7c}	RUNSOL(C)	2
V_{8dc}	DETIN(D,C)	3
V_{9dc}	RUNDEE(D,C)	3
V_{10dc}	DADUST(D,C)	3
$\dot{X}_{0rc}$	AGAINQ(R,C)	
${}_I\dot{X}_{0rc}$	AGAINQ(R,C) [inpart]	4
$\dot{X}_{01r}$	H2QQQQ(R)	
${}_O\dot{X}_{01r}$	H2QQQQ(R) [inpart]	1
X_{21dc}	CLIT(D,C)	3, 5
$\dot{X}_{21dc}$	CLITQQ(D,C)	
${}_C\dot{X}_{21dc}$	CLITQQ(D,C) [inpart]	8, 10
${}_I\dot{X}_{21dc}$	CLITQQ(D,C) [inpart]	3, 4
${}_S\dot{X}_{21dc}$	CLITQQ(D,C) [inpart]	5
${}_S\dot{X}_{24c}$	CORGQQ(D,C) [inpart]	6
X_{24c}	AQUA(C)	2, 7, 9
${}_C\dot{X}_{24c}$	AQUAQQ(C) [inpart]	7, 9
${}_I\dot{X}_{24c}$	AQUAQQ(C [inpart]	2, 4

```
      SUBROUTINE MEDIUM

C DEPTH              THE MEAN DEPTH OF THE SYSTEM IN METERS
C FALL(M)            THE PROPORTION OF THE M'TH CATEGORY OF SUSPENDED
C                    DETRITUS REACHING THE BOTTOM IN A SINGLE TIME UNIT
C COAGUL             THE PROPORTION OF DISSOLVED ORGANIC MATTER COAGULATED
C                    AND HENCE TRANSFERRED TO SUSPENDED DETRITUS IN A
C                    SINGLE TIME UNIT
C-----------------------------------------------------------------------
C      COMMON BLOCK /ACC/ CONTAINS ACCUMULATED CHANGES, WHICH MAY BE
C      NEGATIVE.  COMMON BLOCK /ACCINC/ CONTAINS THE INCREMENTS TO THE
C      ARRAYS IN /ACC/ FOR A SINGLE TIME UNIT.
C-----------------------------------------------------------------------
       COMMON /ACC/ ACAIN(3,5),FROD(3),H2O(7)
       COMMON /ACCINC/ ACAINQ(3,5),FRODQQ(3), H2OQQQ(7)
C-----------------------------------------------------------------------
C      COMMON BLOCK /SPEC/ CONTAINS SPECIFICATIONS AND OTHER INFORMATION
C      COMMON TO THE WHOLE SET OF PROGRAMS, BUT EXCLUDING STATE AND
C      EXOGENOUS VARIABLES.
C-----------------------------------------------------------------------
       COMMON/SPEC/TCOVER,NCHAN,INSTRU(20),    WATER,NSPECV,NSPECA,NORGAN,
     1 NFRACT,NELEM,NOLIT ,NCHECK,IDAY, ATOT, ATOTC,IYRDAY,NREPET(20)
     2,NCOH(20),LISCOH(98),NCOHCU(20),NCOHOP,NFRELM,NFRAC1,NSPCOH,NDEBUG    D18C
     3 ,FLOUT,MICROB, BIOMIN(98),MONTH
C-----------------------------------------------------------------------
C      COMMON BLOCK /STAT/ CONTAINS THE STATE VARIABLES, AND /CHANGE/
C      THEIR INCREMENTS OR DECREMENTS FOR THE CURRENT TIME UNIT.
C-----------------------------------------------------------------------
       COMMON/STAT/ CVEG(20,6,6),CORG(5,6),POP(98),CBIOM(98,6),AQUA(6),     C22C
     1 CLIT(5,6),CRACT(3,6),           DUMMY(98)
       COMMON /CHANGE/ CVEGQQ(20,6,6),CORGQQ(5,6),POPQQ(98),CBIOMQ(98,6)    C2FC
     1,AQUAQQ(6), CLITQQ(5,6),CRACTQ(3,5)            ,DUMMQ(98)
C-----------------------------------------------------------------------
C      COMMON BLOCK /METEOR/ CONTAINS THE VALUES OF EXOGENOUS VARIABLES
C      FOR THE CURRENT TIME UNIT.
C-----------------------------------------------------------------------
       COMMON/METEOR/WIRRIG,ERO,RUNSOL(6),RUNDEB(3,6), DARAIN,DAYRUN,
     1EVAP,WATTEM,DAPHOT, DAYRAD, DADUST(3,6) , EXOG(98), RAINCO(6),
     2COMPIN(6), DETIN(5,6), RUNON,FLOWIN,DRIFTV(20,6,6), DRIFTA(98,6),
     3DRIFTM(3,6), DRIFPO(98)
C-----------------------------------------------------------------------
C      COMMON BLOCK/PARAM/ CONTAINS THE PARAMETERS OF THE SUBROUTINE.
C-----------------------------------------------------------------------
       COMMON/PARAM/ DUMMI(17465),FALL(7), COAGUL
C-----------------------------------------------------------------------
C      THE NAME LIST CONTAINS THE PARAMETERS THAT ARE TO BE
C      READ IN AT EXECUTION TIME.
C-----------------------------------------------------------------------
       NAMELIST/MPUT/ FALL, DEPTH, COAGUL
C-----------------------------------------------------------------------
C      DETRITUS AND ELEMENTS IN SOLUTION ARE GAINED TO THE SYSTEM BY
C      INFLOW AND REMOVED IN THE OUTFLOW (INFLOW-EVAPORATION)
C-----------------------------------------------------------------------
       PRECIP=DARAIN*1000.
       DO 1 J=1,NFRELM
       A=FLOWIN*COMPIN(J)
```

```
      AGAING(2,J)=AGAING(2,J)+A
      AQUAQQ(J)=AQUAQQ(J)+A
      DO 1 I=1,NOLIT
      P=DETIN(I,J)*FLOWIN
      C=DADUST(I,J)
      R=RUNDER(I,J)*RUNON
      CLITQQ(I,J)=CLITQQ(I,J)+P+C+R
      AGAING(1,J)=AGAING(1,J)+C
    1 AGAING(2,J)=AGAING(2,J)+P+R
      FLOUT=AMAX1(0.,FLOWIN-EVAP)
      DO 2 J=1,NFRELM
      S=RUNSOL(J)*RUNON
      IF (AQUA(J).LE.0.) GO TO 12
      A=AQUA(J)*AMIN1(1.,FLOUT*.000001/DEPTH)
      P=RAINCO(J)*PRECIP
      AQUAQQ(J)=AQUAQQ(J)-A+S+P
      AGAING(1,J) = AGAING(1,J) +P
      AGAING(2,J)=AGAING(2,J)-A+S
   12 CONTINUE
      DO 2 I=1,NOLIT
      IF (CLIT(I,J).LE.0.) GO TO 13
      A=CLIT(I,J)*AMIN1(1.,FLOUT*.000001/DEPTH)
      CLITQQ(I,J)=CLITQQ(I,J)-A
      AGAING(2,J)=AGAING(2,J)-A
   13 CONTINUE
    2 CONTINUE
C-----------------------------------------------------------------------
C     THE FOLLOWING DEALS WITH SEDIMENTATION.
C-----------------------------------------------------------------------
      DO 3 I=1,NOLIT
      DO 3 J=1,NFRELM
      IF (CLIT(I,J).LE.0.) GO TO 3
      A=CLIT(I,J)*FALL(I)
      CLITQQ(I,J)=CLITQQ(I,J)-A
      CORGQQ(I,J)=CORGQQ(I,J)+A
    3 CONTINUE
C-----------------------------------------------------------------------
C     THE FOLLOWING DEALS WITH COAGULATION OF DISSOLVED
C     ORGANIC MATTER.
C-----------------------------------------------------------------------
      A=AQUA(1)* COAGUL
      CLITQQ(1,1)=CLITQQ(1,1)+A
      AQUAQQ(1)=AQUAQQ(1)-A
      A = AQUA(5)*COAGUL
      CLITQQ(1,5) = CLITQQ(1,5) + A
      AQUAQQ(5) = AQUAQQ(5) - A
      H2OQQQ(1)=H2OQQQ(1)-EVAP+PRECIP
      H2OQQQ(2)=H2OQQQ(2)+FLOWIN-FLOUT+RUNON
C-----------------------------------------------------------------------
C     THE FOLLOWING ALLOWS FOR THE PRINTING OF DECREMENTS TO
C     THE STATE VARIABLES
C-----------------------------------------------------------------------
      IF(NDEBUG.LE.0) RETURN
      WRITE(6,2007)
 2007 FORMAT(//////)
      WRITE(6,2004)((CLITQQ(I,K),I=1,NOLIT),K=1,NFRELM)
 2004 FORMAT(' CLITQQ    ',10E12.5)
```

```
      WRITE(6,2005)((CORQQQ(I,K),I=1,NCLTT),K=1,NFRELM)
 2005 FORMAT(' CORQQQ     ',10F12.6)
      WRITE(6,2006)(AQUAQQ(K),K=1,NELEM)
 2006 FORMAT(' AQUAQQ     ',10F12.6)
 2008 FORMAT(' AGAINQ     ',10F12.6)
      WRITE (6,2008)   ((AGAINQ(I,J),I=1,7),J=1,5)
      RETURN
      ENTRY MINPUT
      READ (5,MPUT)
      RETURN
      END
```

ANIMAL SUBROUTINE

G. Wayne Minshall
J. Deacon, A. Holman, R. Kramer,
F. Post and C. Stalnaker
J. Wlosinski

INTRODUCTION

This submodel, for use in association with the other general-system aquatic programs of the Desert Biome described elsewhere, covers all processes in which animals are involved -- whether herbivores, carnivores, or detritivores -- and also the processes of the heterotrophic microorganisms.

The animals are divided into species or species groups, and each species group may be divided into different stages of development or cohorts. These subdivisions of the animal biomes are specified at execution time. Similarly, the microorganisms are divided into groups with different substrate relationships (or differing in process rates).

As with the other programs in this series, the ecosystem simulated is regarded as a homogeneous body of water of constant depth, receiving inflow which may carry drifting organisms, and from which outflow may also carry part of the biomass. All quantities in the system are expressed per unit (m^2) of surface area.

The chemical constituents of the biomass modelled may be defined at execution time, but include carbon and chemical energy as a minimum.

VERBAL DESCRIPTION

ANIMALS

The change in biomass of each animal species is compounded of:

- food intake
- respiration
- excretion and egestion
- mortality
- development and reproduction
- drift

with the exception of respiration, all these processes involve all the chemical constituents of the organism.

The rate of food intake per unit biomass (expressed as carbon content) is related to the amount of food availabe. The "food available" is the sum of the carbon per unit area in the various types of food, weighted with a preference factor. The food types to which these preferences may be applied include all plant species groups, all animal cohorts -- and all types of microorganisms -- and of detritus, whether suspended or in the sediments. Consumption of all chemical constituents is proportional to their relative content in the food source, though the weighty factor is applied to the carbon content only.

Of the carbon consumed, in the food, a proportion fixed for the animal cohort in question is assimilated, while other elements are assimilated in such quantities as will keep the chemical composition of the animal unchanged. The remainder is lost as egestion, and enters the suspended detritus.

Respiration rate is a function of water temperature and mean size per individual. Other elements are lost in solution to the water by excretion in proportion to the respiratory loss of carbon.

There is also the possibility of loss of soluble constituents in the animal by leaching, at a rate specific to the animal cohort and the constituent.

Natural mortality takes place at a constant proportional rate specific to the cohort, and dead material being transferred to either suspended or sedimentary detritus, according to whether the cohort is planktonic or benthic in habit.

Migration out of the ecosystem may occur, the rate being dependent on the available food (in the sense above) and the population density.

For transfer of biomass from one cohort to the next cohort of the same species by development, or transfer from the adult category to eggs or young by reproduction, several options are provided.

One option is that the rate of transfer is a function of accumulated mean daily water temperature in excess of a certain threshold, summed since the date when this transfer last reached completion.

An alternative option is where the rate of transfer is an exponential function of the current temperature. At the time of transfer, not all the biomass may enter the more mature cohort. Some (a fixed proportion for that transfer) may be transferred as exuviae to the bottom detritus. If the more mature cohort is a winged form, the rest of the transfer will be lost to the aquatic ecosystem, which is considered to come to an end at the water surface.

Since population adjustments in the cohorts need to be made in accordance with the biomass transfers, the assumption is made that individual biomass within the cohort is distributed uniformly on limited interval and that the largest individuals are transferred.

Reproduction -- biomass transfer into the youngest cohort, either from adults, or from the exterior in the case of flying insects -- may also be controlled in various ways. In some cases, the rate is constant when temperature exceeds a certain threshold; in others the rate is constant within a fixed temperature range; or it may follow a seasonal course, but depending on population density.

MICROORGANISMS

Each type of microorganism distinguished may utilize a particular range of substrates, detritus of various types, or dissolved organic matter with varying ease or difficulty. This is expressed, as in the case of animals, by a preference table. The total potential substrate utilization depends by a Michaelis-Menton function on the weighted total of substrates available, and on the temperature.

It is assumed that the microbial protoplasm is constant in composition, and that if any constituent is in short supply in the substrate it may limit the synthesis of new protoplasm.

The utilization of substrate thus calculated is assumed to be net of losses, whether by respiration, excretion or leaching.

Like the zooplankton, microorganisms are liable to passive drift into and out of the system.

Mathematical Description

<u>Animal feeding and respiration</u>

The rate of increase of animal biomass due to food intake may be expressed as:

$$\sum_s {}_F\dot{X}_{12sc} = - \sum_p {}_H\dot{X}_{1pc} - \sum_s {}_P\dot{X}_{12sc} - \sum_d {}_C\dot{X}_{21dc} - \sum_d {}_C\dot{X}_{22dc} - \sum_m {}_C\dot{X}_{23mc} \qquad (1)$$

where ${}_F\dot{X}_{12sc}$ is the increase of the c'th constituent of the s'th animal cohort due to feeding, while the terms on the left represent the losses from this cause by plants, other animals, suspended detritus, bottom detritus, and heterotrophic microorganisms respectively.

The food intake depends on the sum of food supplies available, weighted by preference factors. Thus,

$$ {}_{F}\dot{X}_{12_{s1}} = P_{1_s} X_{12_{s1}} \{1 - \exp (P_{2_s} Z_{1_s} / Z_{2_s})\} \tag{2}$$

where:

$$ Z_{1_s} = \sum_p P_{3_{sp}} \cdot X_{1_{p1}} + \Sigma\, P_{sb} \cdot X_{12_{b1}} + \sum_d P_{5_{sd}} X_{21_{d1}} + \sum_d P_{6_{sd}} X_{22_{d1}} + \Sigma\, P_{7_{sm}} X_{23_{m1}} \tag{3}$$

the denominator being the sum of the weights:

$$ Z_{2_s} = \sum_p P_{3_{sp}} + \sum_b P_{4_{sb}} + \sum_d P_{5_{sd}} + \sum_d P_{6_{sd}} + \sum_m P_{7_{sm}} \tag{4}$$

In these expressions, $P_{3_{sp}}$ is the preference factor of the s'th animal cohort for food from the p'th plant species group, while $P_{4_{sb}}$, $P_{5_{sd}}$, $P_{6_{sd}}$ and $P_{7_{sm}}$ similarly express preferences for food from the b'th animal cohort, the d'th type of heterotrophic microorganisms respectively.

Correspondingly, the loss of carbon from tissue of the p'th plant group due to herbivory by the s'th animal cohort is

$$ Z_{3_{ps1}} = P_{3_{sp}} X_{1_{p1}} Z_{2_s} / Z_{1_s} \tag{5}$$

and similarly losses from other animal cohorts, detritus and microorganisms through consumption by the s'th animal cohort are:

$$ Z_{4_{bs1}} = P_{4_{sb}} X_{11_{b1}} Z_{25} / Z_{1_s} \tag{6}$$

$$ Z_{5_{ds1}} = P_{5_{sd}} X_{21_{d1}} Z_{25} / Z_{1_s} \tag{7}$$

$$ Z_{6_{ds1}} = P_{6_{sd}} X_{22_{d1}} Z_{25} / Z_{1_s} \tag{8}$$

$$ Z_{7_{ms1}} = P_{7_{sm}} X_{23_{m1}} Z_{25} / Z_{1_s} \tag{9}$$

Losses of other elements from the various food categories are proportional to the loss in carbon, thus,

$$ Z_{3_{psc}} = Z_{3_{ps1}} X_{1_{pc}} / X_{1_{p1}} \tag{10}$$

$$ Z_{4_{bsc}} = Z_{4_{bs1}} X_{12_{bc}} / X_{12_{b1}} \tag{11}$$

$$ Z_{5_{dsc}} = Z_{5_{ds1}} X_{21_{dc}} / X_{21_{d1}} \tag{12}$$

$$ Z_{6_{dsc}} = Z_{6_{ds1}} X_{22_{dc}} / X_{22_{d1}} \tag{13}$$

The proportion of the food carbon intake assimilated is constant for each animal cohort, thus:

$$ {}_{A}\dot{X}_{12_{s1}} = P_{8_s}\, {}_{F}\dot{X}_{12_{s1}} \tag{15} $$

The proportional increase of other elements in the biomass due to assimilation of food is the same as that for carbon:

$$ {}_{A}\dot{X}_{12_{sc}} = {}_{A}\dot{X}_{12_{s1}} \cdot X_{12_{sc}} / X_{12_{s1}} \tag{16} $$

For the individual, the relation between the logarithm of the respiration rate and the logarithm of biomass is linear. Assuming a constant Q_{10}, this is,

$$ Z_{8_s} = P_{9_s} + P_{10_s}(X_{12} - X_{11}) + P_{11_s} V_1 \tag{17} $$

where Z_{8_s} is the logarithm of the individual respiration rate, V_1 is the water temperature, P_{9_s}, P_{10_s}, and P_{11_s} are constants, and the other terms have the meanings already defined. Then the respiration rate for the whole biomass of this cohort is

$$ {}_{R}\dot{X}_{12_{s1}} = X_{11_s} \cdot \exp(Z_{8_s}) \tag{18} $$

Basic metabolism results both in respiration and excretion; so, as an approximation, the whole of the respiratory loss of carbon is matched by a similar proportional loss of each of the other constituents:

$$ {}_{R}\dot{X}_{12_{sc}} = \dot{X}_{12_{s1}} \cdot X_{12_{sc}} / X_{12_{s1}} \tag{19} $$

The balance between food intake and the sum of assimilation and respiration is exgested so that:

$$ {}_{E}\dot{X}_{12_{sc}} = {}_{F}\dot{X}_{12_{sc}} - {}_{A}\dot{X}_{12_{sc}} - {}_{R}\dot{X}_{12_{sc}} \tag{20} $$

Excreted carbon goes into suspended detritus

$$ {}_{E}\dot{X}_{21_{11}} = -\sum_{s} {}_{E}\dot{X}_{12_{s1}} \tag{21} $$

while respired carbon goes into dissolved bicarbonate, and thence to the atmosphere:

$$ {}_{R}\dot{X}_{0_{11}} = -\sum_{s} {}_{R}\dot{X}_{12_{s1}} \tag{22} $$

Micro-organism substrate use

Use of substrates by micro-organisms is controlled by preference tables similar to those for animals. The change in the c'th microbial constituent by assimilation is

$$\sum_m {}_A\dot{X}_{23_{mc}} = -\sum_d {}_M\dot{X}_{21_{dc}} - \sum_d {}_M\dot{X}_{22_{dc}} - {}_M\dot{X}_{24_c} \tag{23}$$

Where ${}_A\dot{X}_{23_{mc}}$ is the increase in the m'th type of microorganisms, and the terms on the left represent the losses from this cause by suspended and bottom detritus, and by dissolved material, respectively.

The potential intake depends on the temperature and on the sum of substrates available, weighted by preference factors, thus,

$$Z_{8_{m1}} = \exp\,(P_{12} + P_{13}V_1 + P_{14}V_1)/(1 + P_{15_m} Z_{10_m} / Z_{9_m}) \tag{24}$$

where the denominator of the last fraction expressing substrate availability is the weighted sum of the various substrates

$$Z_{9_m} = \sum_d P_{16_{md}} X_{21_{d1}} + \sum_d P_{17_{md}} X_{22_{d1}} + P_{18_m} X_{24_1} \tag{25}$$

while the numerator is the sum of the weights

$$Z_{10_m} = \sum_d P_{16_{md}} \quad \sum_d P_{17_{md}} + P_{18_m} \tag{26}$$

In these expressions, $P_{16_{md}}$ is the preference factor by the m'th microbial type for the d'th type of suspended detritus, while $P_{17_{md}}$ and P_{18_m} similarly express preferences for bottom detritus and for dissolved material.

Correspondingly, the potential loss of carbon from the d'th type of suspended detritus due to action by the m'th microbial type is

$$Z_{11_{md1}} = P_{16_{md}} X_{21_{d1}} Z_{10_m} / Z_{9_m} \tag{27}$$

and losses from bottom detritus and dissolved material are:

$$Z_{12_{md1}} = P_{17_{md}} X_{22_{d1}} Z_{10_m} / Z_{9_m} \tag{28}$$

$$Z_{13_{m1}} + P_{18_m} X_{24_1} Z_{10_m} / Z_{9_m} \tag{29}$$

These potential changes attain actuality only if the supply of other elements in the substrates is adequate to supply the necessary dietary balance for the microorganisms.

The potential loss of carbon, $Z_{11_{md1}}$ would entail a corresponding loss of the e'th element:

$$Z_{11_{mde}} = Z_{11_{md1}} \cdot X_{23_{me}} / X_{23_{m1}} \tag{30}$$

The actual loss of each element is then determined by whichever element is in minimum supply in the substrate, relative to the bacterial needs. If

$$Z_{14_{md}} = \min\{1, \min_{e} (X_{21_{de}} / Z_{11_{mde}})\} \tag{31}$$

then:

$$_{M}\dot{X}_{21_{de}} = \sum_{m} Z_{14_{md}} Z_{11_{mde}} \tag{32}$$

And similarly:

$$Z_{15_{md}} = \min\{1, \min_{e} (X_{22_{de}} / Z_{12_{mde}})\} \tag{33}$$

$$Z_{16_{m}} = \min\{1, \min_{e} (X_{24_{e}} / Z_{13_{me}})\} \tag{34}$$

$$_{M}\dot{X}_{22_{de}} = \sum_{m} Z_{15_{md}} Z_{12_{mde}} \tag{35}$$

$$_{M}\dot{X}_{24_{e}} = \sum_{m} Z_{16_{m}} Z_{13_{me}} \tag{36}$$

The gain in the e'th constituent by the m'th microbial type is:

$$_{M}\dot{X}_{23_{me}} = \sum_{d} Z_{14_{md}} Z_{11_{mde}} + \sum_{d} Z_{15_{md}} Z_{12_{mde}} + Z_{16_{m}} Z_{13_{me}} \tag{37}$$

Animals may lose constituents to the water by leaching or exudation at a constant rate:

$$_{L}\dot{X}_{12_{se}} = P_{19_{se}} X_{12_{se}} \tag{38}$$

$$_{L}\dot{X}_{24_{e}} = - \sum_{s} {_{L}\dot{X}_{12_{se}}} \tag{39}$$

Natural mortality is also considered to take place at a constant rate, so

$$_{M}\dot{X}_{12_{se}} = P_{20_{s}} X_{12_{se}} \tag{40}$$

$$_{M}\dot{X}_{11_{s}} = P_{20_{s}} X_{11_{s}} \tag{41}$$

In planktonic organisms, the dead bodies become suspended detritus:

$$ {}_{D}\dot{X}_{21_{1c}} = -\sum_{s\epsilon P} {}_{M}\dot{X}_{12_{sc}} \tag{42} $$

where P is defined as the set of animal cohorts with a planktonic habit, and

$$ {}_{D}\dot{X}_{22_{sc}} = -\sum_{s\epsilon B} {}_{M}\dot{X}_{12_{sc}} \tag{43} $$

where B is the set of benthic cohorts.

The proportion of animal populations migrating out of the area is a linear function of the available food and of the populaiton density:

$$ Z_{17_s} = P_{21_s} + P_{22_s} Z_{1_s} / Z_{2_s} + P_{23_s} X_{11_s} \tag{44} $$

It is assumed that the migrating individuals are a random sample of the population,

$$ {}_{S}\dot{X}_{11_s} = Z_{17_s} X_{11_s} \tag{45} $$

$$ {}_{S}\dot{X}_{12_{sc}} = Z_{17_s} X_{12_{sc}} \tag{46} $$

The resulting loss to the ecosystem as a whole is expressed:

$$ {}_{S}\dot{X}_{0_c} = \sum_{s} {}_{S}\dot{X}_{12_{sc}} \tag{47} $$

Where an animal population consists of more than one sub-population (cohort) at different stages of development, transfer of individuals from one stage to another is determined in various ways, which may be special to each animal group and cohort.

It will be convenient to designate the cohort currently under consideration as s; that <u>into</u> which material may be transferred from s will be r, that <u>from</u> which material may be received by s will be t.

Most commonly $s = r - 1 = t + 1$; usually the relationship between s and t is one-to-one, but sometimes several different cohorts may transfer material into the same cohort (e.g. when reproduction takes place in several cohorts), or a single cohort may transfer into more than one (as when some individuals oviparous and others are viviparous). In these cases s or t must be considered as a set of cohorts, and some of the functions to be described will need summation over the set.

A number of alternative functions are used to describe the rates of transfer, and the sets of cohorts to which these apply will be designated S_1 S_2.....

In most cases the rate of transfer from cohort t to cohort s is a function of accumulated water temperature V_{2_S}, summed since the date when this transfer last reached completion:

$$Z_{18_s} = \max\ [0, \min\ \{1, (V_{2_s} - P_{24_s})/P_{25_s}\}],\ s \varepsilon S_1 \tag{48}$$

In other cases, the rate of transfer is an exponential function of the current temperature

$$Z_{18_s} = \max\ [0, \min\ \{1, \{P_{26_s} + P_{27_s} \exp\ (P_{28_s} V_1)\}\}],\ s \varepsilon S_2 \tag{49}$$

The biomass transferred out of cohort t is often not all transferred into cohort s, for some may be lost as exuviae (eggshells, cast skins, etc). The proportion of material so lost -- assumed the same for all chemical constituents -- is P_{29_t}. Thus, the change in the c'th biomass component of the s'th cohort due to development processes, is:

$$_D\dot{X}_{12_{sc}} = Z_{18_s} \cdot X_{12_{tc}} \cdot (1 - P_{29_t}) - Z_{18_r} X_{12_{sc}} \tag{50}$$

If the s'th cohort is a flying form, it leaves the system immediately:

$$_F\dot{X}_{2_{sc}} = - {_D\dot{X}_{12_{sc}}},\quad s \varepsilon S_3 \tag{51}$$

and:

$$_F\dot{X}_{0_c} = \sum_{s \varepsilon S_3} {_D\dot{X}_{12_{sc}}} \tag{52}$$

The exuviae enter the bottom detritus

$$_D\dot{X}_{22_{2c}} = Z_{18_s} X_{12_{tc}} P_{29_t} \tag{53}$$

The changes in population are calculated on the assumptions that individual biomass is uniformly distributed on an interval of which the lowest value (in terms of carbon content) is recorded as X_{13_s}, and that the largest individuals are those which are transferred.

The carbon content in individuals transferred from the t'th to the s'th cohort then ranges from:

$$(2X_{12_{tl}}/X_{11_t} - X_{13_t})\ (1 - Z_{18_s}) + Z_{18_s}X_{13_t}$$

to

$$2X_{12_{tl}}/X_{11_t} - X_{13_t}$$

To mean carbon content over this range is

$$1/2\ \{(2X_{12_{tl}}/X_{11_t} - X_{13_t})\ (2 - Z_{18_s}) + Z_{18_s}X_{13_t}$$

and the population transfer will consequently be:

$$Z_{19_s} = Z_{18_s}/\{2\ X_{12_{tl}} - X_{11_t} - X_{13_t} - Z_{18_s}(X_{12_t}/X_{11_t} - X_{13_t})\} \tag{54}$$

Thus:

$$_D\dot{X}_{11_s} = Z_{19_s} - Z_{19_r} \tag{55}$$

Where the adults have an aerial habit ($s\varepsilon S_3$), the transfer into the egg cohort is an exogeneous variable - V_{3_r} being the number of eggs laid per unit water area per time unit, and:

$$_R\dot{X}_{11_r} = V_3 \tag{56}$$

$$_R\dot{X}_{12_{rc}} = P_{30_{rc}}V_{3_r} \tag{57}$$

where $P_{30_{rc}}$ is the content per egg (r'th cohort) of the c'th constituent.

Where the adults remain within the system, transfers of biomass in the course of reproduction may be expressed by various functions specific to the different animal groups. Where s is again the receiving cohort (normally eggs), these transfer rates may be expressed

$$\left.\begin{array}{ll} Z_{18_s} = P_{31_s} & ,\quad V_1 \geq P_{32_s} \\ Z_{18_s} = 0 & ,\quad V_1 \leq P_{32_s} \end{array}\right] s\varepsilon S_4 \tag{58}$$

when the rate is constant if the temperature exceeds the threshold value P_{3_s};

$$\left.\begin{array}{ll} Z_{18_s} = P_{33_s} & ,\quad P_{32_s} \leq V_1 \leq P_{34_s} \\ Z_{18_s} = 0 & ,\quad V_1 < P \ \text{or}\ V_1 > P \end{array}\right] s\varepsilon S_5 \tag{59}$$

where a constant rate applies within a specified temperature range; or

$$Z_{18_s} = P_{35_s}\{P_{36_s} + P_{37_s}\sin(V_4 + P_{38_s})\}\,[\min\{1, \max(0, P_{39_s} + P_{40_s}\ln(X_{11_t})\}] \quad (60)$$

There V_4 is the Julian day, so that the rate follows a periodic seasonal course provided the temperature is within an acceptable range, and also depends on population density.

The changes through reproduction in the s'th cohort are then:

$$_R\dot{X}_{12_{sc}} = Z_{18_s} X_{12_{tc}} \quad (61)$$

$$_R\dot{X}_{11_s} = \dot{X}_{12_{sl}}/P_{41_s} \quad (62)$$

$$_R\dot{X}_{12_{tc}} = Z_{18_s} X_{12_{tc}} \quad (63)$$

where P_{41_s} is the mean weight of carbon in a new individual of the s'th cohort.

The proportion of the population of the s'th cohort exposed to loss by drift is in some benthic organisms dependent on a relation between food supply and population density:

$$Z_{19_s} = \max\{0, \min[1, \{P_{42} + R_{43}\ln(X_{11_s}) + P_{44}(\ln X_{11_s})^2 - P_{45}(\ln Z_{1_s}/Z_{2_s})]$$

$$, \quad s\varepsilon D_1 \quad (64)$$

in other benthic organisms there is no risk of drift:

$$Z_{20_s} = 0 \quad , \quad s\varepsilon D_2 \quad (65)$$

while for the plankton, the whole population is exposed to it:

$$Z_{20_s} = 1 \quad , \quad s\varepsilon D_4 \quad (66)$$

If the volume of the system is V_5 and the rate of outflow $_O X_{01_2}$, the losses by drift are then

$$_O\dot{X}_{12_s} = -\exp({_O\dot{X}_{01_2}}/V_5)\, X_{12_{sc}} Z_{20_s} \quad (67)$$

$$_O\dot{X}_{0_c} = -{_O\dot{X}_{12_{sc}}} \quad (68)$$

$$_O\dot{X}_{11_s} = -\exp({_O\dot{X}_{01_2}}/V_5)\, X_{11_{sc}} Z_{20_s} \quad (69)$$

Table of symbols

Symbol	FORTRAN	Equation
P_{1_s}	TAKE(S)	2
P_{2_s}	CURVE(S)	2
$P_{3_{sp}}$	PREFV(S,P)	3, 4, 5
$P_{4_{sb}}$	PREFA(S,B)	3, 4, 6
$P_{5_{sd}}$	PREFL(S,D)	3, 4, 7
$P_{6_{sd}}$	PREFO(S,D)	3, 4, 8
$P_{7_{sm}}$	PREFM(S,M)	3, 4, 9
P_{8_s}	ASSIM(S)	15
P_{9_s}	SLOPE(S)	16
P_{10_s}	CONST(S)	16
P_{10}	CONTA	21
P_{11}	CONTB	21
P_{12}	CONTC	21
P_{13_m}	CONS(M)	21
$P_{14_{md}}$	PREFER(M,D)	22, 23, 24
$P_{15_{md}}$	PREFER(M,D)	22, 23, 25
P_{16_m}	PREFER(M)	22, 23, 26
$P_{17_{sc}}$	SIEVE(S,C)	35
P_{18_s}	AMORTA(S)	37, 38
P_{19_s}	AMIG3(S)	41
P_{20_s}	AMIG1(S)	41
P_{21_s}	AMIG2(S)	41
P_{22_s}	THRES(S)	45
P_{23_s}	AMAX(S)	45
P_{24_s}	HATCON(S)	46
P_{25_s}	TEMCON(S)	46
P_{26_s}	HATCOB(S)	46
P_{27_t}	SHELP(T)	47, 50

Continued

Table of symbols, continued

Symbol	FORTRAN	Equation
$P_{28_{rc}}$	EGCOMP(R,C)	57
$P_{29_{s}}$	REPROD(S)	58
$P_{30_{s}}$	TRESH(S)	58, 59, 60, 61
$P_{31_{s}}$	RCONST(S)	59, 60
$P_{32_{s}}$	UPTHRE(S)	61
$P_{33_{s}}$	CONSTA(S)	62
$P_{34_{s}}$	CONSTC(S)	62
$P_{35_{s}}$	CONSTB(S)	62
$P_{36_{s}}$	CONSA(S)	62
$P_{37_{s}}$	CONSB(S)	62
$P_{38_{s}}$	BIOMIN(S)	66
$P_{39_{s}}$	CONPAR(S)	67
$P_{40_{s}}$	CONBAR(S)	67
$P_{41_{s}}$	CONCAR(S)	67
$P_{42_{s}}$	CONDAR(S)	67
${}_{F}\dot{X}_{12_{sc}}$	CBIOMA(S,C)[in part]	1
${}_{H}\dot{X}_{1_{pc}}$	CVEGQQ(P,1,C)[in part]	1
${}_{P}\dot{X}_{12_{sc}}$	CBIOMA(S,C)[in part]	1
${}_{C}\dot{X}_{21_{dc}}$	CLITQQ(D,C)[in part]	1
${}_{C}\dot{X}_{22_{dc}}$	CORGQQ(D,C)[in part]	1
${}_{C}\dot{X}_{23_{mc}}$	CBACTQ(M,C)[in part]	1
$X_{12_{s1}}$	CBIOM(S,1)	2, 15
$X_{1_{p1}}$	CVEG(P,1)	3, 5, 10
$X_{12_{b1}}$	CBIOM(B,1)	3, 6, 11
$X_{21_{d1}}$	CLIT(D,1)	3, 7, 12, 22, 24
$X_{22_{d1}}$	CORG(D,1)	3, 8, 13, 22, 25
$X_{23_{mc}}$	CBACT(M,1)	3, 9, 14

Continued

Table of symbols, continued

Symbol	FORTRAN	Equation
X_{1pc}	CVEG(P,1,C)	10
X_{12bc}	CBIOM(B,C)	11
X_{21dc}	CLIT(D,C)	12
X_{22dc}	CORG(D,C)	13
X_{23mc}	CBACT(M,C)	14
${}_{A}X_{12s1}$	CBIOMQ(S,1)[in part]	15
${}_{F}\dot{X}_{12s1}$	CBIOM(S,1)[in part]	15, 16
${}_{R}\dot{X}_{12s1}$	CBIOMQ(S,1)[in part]	16
${}_{E}\dot{X}_{12sc}$	CBIOMQ(S,C)[in part]	17
${}_{P}\dot{X}_{12sc}$	CBIOMQ(S,C)[in part]	17
${}_{E}\dot{X}_{21d1}$	CUTQQ(C,1)[in part]	18
${}_{R}\dot{X}_{321}$	AQUAQQ(1)[in part]	19
${}_{A}\dot{X}_{23mc}$	CBACTQ(M,C)[in part]	20
${}_{M}\dot{X}_{21dc}$	CLITQQ(D,C)[in part]	20, 29
${}_{M}\dot{X}_{22dc}$	CORGQQ(D,C)[in part]	20, 32
${}_{M}\dot{X}_{24dc}$	AQUAQQ(D,C)[in part]	20, 33
X_{241}	AQUA(1)	22, 26
${}_{M}\dot{X}_{23mc}$	CBACTO(M,C)[in part]	34
${}_{L}\dot{X}_{12sc}$	CBIOMQ(S,C)[in part]	35, 36
${}_{L}\dot{X}_{24c}$	AQUAQQ(C)[in part]	36
X_{12sc}	CBIOM(S,C)	35, 37, 43
${}_{M}\dot{X}_{12sc}$	CBIOMQ(S,C)[in part]	37, 39, 40
${}_{M}\dot{X}_{11s}$	POPQQQ(S)[in part]	38
${}_{D}\dot{X}_{21sc}$	CLITQQ(1,C)[in part]	39
${}_{D}\dot{X}_{22sc}$	CORGQQ(S,C)[in part)	40
X_{11s}	POP(S)	41, 42
${}_{S}\dot{X}_{11s}$	POPQQQ(S)[in part]	42

Continued

Table of symbols continued

Symbol	FORTRAN	Equation
${}_{S}\dot{X}_{12_{sc}}$	CBIOMQ(S)[in part]	43, 44
${}_{S}\dot{X}_{0_{sc}}$	AGAINQ(S,C)[in part]	44
${}_{D}\dot{X}_{12_{sc}}$	CBIOMQ(S,C)[in part]	47, 48
${}_{F}\dot{X}_{12_{sc}}$	CBIOMQ(S,C)[in part]	48
${}_{F}\dot{X}_{0_{sc}}$	AGAINQ(S,C)[in part]	49
${}_{D}\dot{X}_{22_{sc}}$	CORGQQ(2,C)[in part]	50
$X_{12_{t2}}$	CBIOM(T,1)	51, 53, 54
$X_{11_{t}}$	POP(T)	51, 52, 53, 54, 62, 67
$X_{13_{t}}$	BIOMIN(T)	51, 52, 53, 54
${}_{D}\dot{X}_{11_{s}}$	POPQQQ(S)[in part]	55
${}_{R}\dot{X}_{11_{r}}$	POPQQQ(S)[in part]	56, 65
${}_{R}\dot{X}_{12_{rc}}$	CBIOMQ(R,C)[in part]	57
${}_{R}\dot{X}_{12_{sc}}$	CBIOMQ(S,C)[in part]	64
${}_{R}\dot{X}_{12_{tc}}$	CBIOMQ(T,C)[in part]	66
$X_{12_{sc}}$	CBIOM(S,C)	70
${}_{O}\dot{X}_{11_{s}}$	POPQQQ(S)[in part]	70
${}_{O}\dot{X}_{0_{sc}}$	AGAINQ(S,C)[in part]	71
${}_{O}\dot{X}_{12_{sc}}$	CBIOMQ(S,C) [in part]	71
$Z_{1_{s}}$	SUM	2,3,5,6,7,8,9,41,67
$Z_{2_{s}}$	TOT	2,4,5,41,67
$Z_{3_{ps1}}$	TAKING	5
$Z_{4_{bs1}}$	TAKING	6
$Z_{5_{ds1}}$	TAKING	7
$Z_{6_{ds1}}$	TAKING	8
$Z_{7_{ms1}}$	TAKING	9
$Z_{3_{psc}}$	FACTOR	10
$Z_{4_{bsc}}$	FACTOR	11

Continued

Table of symbols, continued

Symbol	FORTRAN	Equation
$Z_{5_{dsc}}$	FACTOR	12
$Z_{6_{dsc}}$	FACTOR	13
$Z_{7_{msc}}$	FACTOR	14
$Z_{8_{m1}}$	B	21
$Z_{9_{m}}$	ATE(I)	21,22,24,25,26
$Z_{10_{m}}$	A	21,23,24,25,26
$Z_{11_{md1}}$	SUBSTQ(M,D)	24,27,34
$Z_{12_{md1}}$	SUBSTQ(M,D)	25,34
$Z_{13_{m1}}$	SUBSTQ(M)	26,34
$Z_{14_{md}}$	SUBST(M,D)	28,29,34
$Z_{14_{md}}$	SUBST(M,D)	30,32,34
$Z_{16_{m}}$	SUBST(M,D)	31,33,34
$Z_{16_{s}}$	AMIGRA	41,42,43
$Z_{17_{s}}$	HATCH	45,46,47,50,51,53,54, 58,59,60,61,62,63,64,66
$Z_{18_{s}}$	CHANGE(S)	54,55
$Z_{19_{s}}$	DROUT(S)	67,68,69,70
V_{1}	WATTEM	16
V_{2}	ACCUM	45
$V_{3_{r}}$	EXOGEN(R)	56
V_{4}	IYRDAY	62
V_{5}	DEPTH	70

Continued

```
      SUBROUTINE ANIMAL                                              0020

C ACCUM(L)           THE ACCUMULATED TEMPERATURES FOR CHANGES IN THE L'TH
C                    ANIMAL COHORT
C AMAX(I)            VALUE OF ACCUMULATED TEMPERATURES AT WHICH THE WHOLE
C                    OF THE (I-1)'TH ANIMAL COHORT IS TRANSFERRED TO THE
C                    L'TH COHORT
C AMIGRA             PROPORTION OF THE ANIMAL COHORT CURRENTLY UNDER
C                    CONSIDERATION MIGRATING OUT OF THE SYSTEM
C AMIG1(L)           COEFFICIENT OF DEPENDENCE OF MIGRATION BY THE L'TH
C                    ANIMAL COHORT ON FOOD SUPPLY
C AMIG2(L)           COEFFICIENT OF DEPENDENCE OF MIGRATION BY THE L'TH
C                    ANIMAL COHORT ON POPULATION DENSITY
C AMIG3(L)           CONSTANT TERM IN MIGRATION FUNCTION FOR THE L'TH
C                    ANIMAL COHORT (PROBABILITY OF MIGRATION WHEN BOTH
C                    FOOD SUPPLY AND POPULATION ARE VERY LOW)
C AMORTA(L)          PROPORTION OF INDIVIDUALS OF THE L'TH ANIMAL COHORT
C                    DYING IN A TIME UNIT
C ASSI               QUANTITY OF FOOD ASSIMILATED PER BIOMASS  OF CARBON
C                    PER TIME UNIT BY THE ANIMAL COHORT CURRENTLY UNDER
C                    CONSIDERATION
C ASSIM(L)           PROPORTION OF FOOD ASSIMILATED BY THE L'TH ANIMAL COHORT
C ATE(T)             SUM OF FOOD PREFERENCE WEIGHTS OF THE T'TH GROUP
C                    OF HETEROTROPHIC MICRO ORGANISMS
C BIONEX             BIOMASS OF THE SMALLEST INDIVIDUAL IN A TRANSFER
C                    BETWEEN ANIMAL COHORTS
C CHELEM             QUANTITY  PER UNIT AREA  OF AN ELEMENT TRANSFERRED
C                    FROM ADULTS TO EGGS DURING REPRODUCTION
C CHANGE             NUMBERS  OR PROPORTION OF BIOMASS   TRANSFERRED INTO
C                    THE ANIMAL COHORT CURRENTLY UNDER CONSIDERATION
C CONBAR(L)          LINEAR TERM IN EXPRESSION RELATING THE PROPORTION
C                    OF THE L'TH ANIMAL COHORT LIABLE TO DRIFT TO THE
C                    POPULATION DENSITY
C CONCAR(L)          QUADRATIC TERM IN EXPRESSION RELATING THE PROPORTION
C                    OF THE L'TH ANIMAL COHORT LIABLE TO DRIFT TO THE
C                    POPULATION DENSITY
C CONDAR(M)          CONSTANT DETERMINING EFFECT OF FOOD SUPPLY ON DRIFT
C                    IN THE M'TH ANIMAL COHORT
C CONPAR(L)          CONSTANT TERM IN EXPRESSION FOR DEPENDENCE OF THE
C                    PROPORTION OF THE L'TH ANIMAL COHORT LIABLE TO DRIFT
C                    ON POPULATION DENSITY
C CONS(T)            MICHAELIS CONSTANT FOR RATE OF SUBSTRATE USE BY THE
C                    T'TH GROUP OF HETEROTROPHIC MICROORGANISMS
C CONSA(L)           CONSTANT TERM IN EXPRESSION FOR DEPENDENCE OF
C                    REPRODUCTION ON POPULATION DENSITY IN THE L'TH
C                    ANIMAL COHORT
C CONSB(L)           COEFFICIENT OF DEPENDENCE OF REPRODUCTION OF THE L'TH
C                    ANIMAL COHORT ON LOG (POPULATION DENSITY)
C CONST(L)           RESPIRATION (G. OF CARBON) PER TIME UNIT AT ZERO
C                    TEMPERATURE BY AN INDIVIDUAL OF THE L'TH ANIMAL
C                    COHORT CONTAINING ONE GRAM OF CARBON
C CONTA              CONSTANT TERM FOR TEMPERATURE DEPENDENCE OF SUBSTRATE
C                    USE BY HETEROTROPHIC MICROORGANISMS (I.E. LOG
C                    (MAXIMUM RATE) AT ZERO TEMPERATURE)
C CONTB              LINEAR COEFFICIENT FOR TEMPERATURE DEPENDENCE OF
C                    SUBSTRATE USE BY HETEROTROPHIC MICRO-ORGANISMS
```

```
C CONTC            QUADRATIC COEFFICIENT FOR TEMPERATURE DEPENDENCE OF
C                  SUBSTRATE USE BY HETEROTROPHIC MICRO-ORGANISMS
C CONSTA(L)        CONSTANT TERM IN EXPRESSION FOR SEASONAL DEPENDENCE
C                  OF OVIPOSITION RATE
C CONSTB(L)        DISPLACEMENT TERM IN SINE FUNCTION FOR SEASONAL
C                  DEPENDENCE OF OVIPOSITION RATE
C CONSTC(L)        AMPLITUDE TERM IN SINE FUNCTION FOR SEASONAL
C                  DEPENDENCE OF OVIPOSITION RATE
C CURVE(L)         FACTOR RELATING FOOD INTAKE IN THE L'TH ANIMAL
C                  COHORT FOOD SUPPLY
C DROUT            BIOMASS (OR NUMBERS) OF THE ANIMAL COHORT CURRENTLY
C                  UNDER CONSIDERATION LOST TO THE ECOSYSTEM BY DRIFT
C DUMMY1(I)        DUMMY ARRAY IN THE PARAM COMMON BLOCK USED BY AN
C                  OTHER SUBROUTINE
C DUMMY2(2)        DUMMY ARRAY IN THE PARAM COMMON BLOCK USED BY ANOTHER
C                  SUBROUTINE
C EGCOMP(K,N)      THE AMOUNT OF THE N'TH CHEMICAL CONSTITUENT IN AN EGG
C                  OF THE K'TH ANIMAL SPECIES
C EXCR             RATE OF EXCRETION PER UNIT BIOMASS (OF CARBON) PER
C                  TIME UNIT FOR THE ANIMAL COHORT CURRENTLY UNDER
C                  CONSIDERATION
C EXOGEN(L)        NUMBERS (PER SQ. M.) OF INDIVIDUALS OF THE L'TH
C                  ANIMAL COHORT ENTERING THE SYSTEM FROM ABOVE (E.G.
C                  OVIPOSITION)
C FACTOR           PROPORTION OF A PARTICULAR FOOD TYPE TO BE CONSUMED
C                  BY THE ANIMAL COHORT CURRENTLY UNDER CONSIDERATION.
C                  ALSO, PROPORTION OF WATER MASS INVOLVED IN OUTFLOW
C FEEDA(L)         QUANTITY OF THE L'TH ANIMAL COHORT WEIGHTED BY THE
C                  DIETARY PREFERENCE OF THE ANIMAL COHORT CURRENTLY
C                  UNDER CONSIDERATION
C FEEDS(M)         QUANTITY OF THE M'TH TYPE OF SUSPENDED DETRITUS,
C                  WEIGHTED BY THE DIETARY PREFERENCE OF THE ANIMAL
C                  SPECIES GROUP CURRENTLY UNDER CONSIDERATION
C FEEDM(I)         QUANTITY OF THE I'TH TYPE OF HETEROTROPHIC
C                  MICROORGANISMS, WEIGHTED BY THE DIETARY PREFERENCE
C                  OF THE ANIMAL SPECIES GROUP CURRENTLY UNDER
C                  CONSIDERATION
C FEEDB(M)         QUANTITY OF THE M'TH TYPE OF BOTTOM DETRITUS,
C                  WEIGHTED BY THE DIETARY PREFERENCE OF THE ANIMAL
C                  SPECIES GROUP CURRENTLY UNDER CONSIDERATION
C FEEDV(I)         QUANTITY OF THE I'TH PLANT SPECIES GROUP, WEIGHTED BY
C                  THE DIETARY PREFERENCE OF THE ANIMAL SPECIES GROUP
C                  CURRENTLY UNDER CONSIDERATION
C FOOD(K)          FOOD INTAKE BY THE K'TH ANIMAL SPECIES GROUP, PER
C                  UNIT OF BIOMASS
C FRACT            RECIPROCAL OF USE OF A PARTICULAR SUBSTRATE BY THE
C                  GROUP OF HETEROTROPHIC MICROORGANISMS CURRENTLY
C                  CONSIDERED, IF UNLIMITED BY ANY CONSTITUENT
C HATCH            PROPORTION OF BIOMASS OR INDIVIDUALS OF ANOTHER
C                  COHORT TO BE TRANSFERRED INTO THE ANIMAL COHORT
C                  UNDER CONSIDERATION
C HATCOB(L)        CONSTANT TERM IN EXPRESSION FOR RATE OF TRANSFER
C                  INTO THE L'TH ANIMAL COHORT
C HATCON(L)        MULTIPLIER FOR TEMPERATURE DEPENDENCE OF RATE OF
C                  TRANSFER INTO THE L'TH ANIMAL COHORT
C IAFATE(L)        THE FATE OF DEAD BODIES OF THE L'TH ANIMAL COHORT
C                  (0 INDICATES THAT THEY BECOME SUSPENDED DETRITUS,
```

```
C                         ( BOTTOM DET  TUS)
C KDAY                    THE DAY ON WHICH TH  SUBROUTINE WAS LAST CALLED
C NDRIFA(L)               SPECIFICATION OF THE LIABILITY OF THE L'TH ANIMAL
C                         COHORT TO DRIFT OUT OF THE SYSTEM
C NTRANS(L)               INDICATES THE CONTROLLING MECHANISM FOR TRANSFER
C                         FROM THE (L-1)'TH TO THE L'TH ANIMAL COHORT
C PARAB                   FUNCTION EXPRESSING DEPENDENCE OF DRIFT ON
C                         POPULATION DENSITY AND FOOD AVAILABILITY
C POPFUN                  FUNCTION EXPRESSING DEPENDENCE OF REPRODUCTION ON
C                         POPULATION DENSITY
C PREFA(K,KK)             PREFERENCE FACTOR FOR CONSUMPTION OF THE KK'TH ANIMAL
C                         SPECIES GROUP BY THE K'TH ANIMAL SPECIES GROUP
C PREFB(T,S)              PREFERENCE FACTOR FOR CONSUMPTION OF THE S'TH TYPE
C                         OF SUBSTRATE (DETRITUS OR DISSOLVED MATERIAL) BY THE
C                         THE T'TH GROUP OF HETEROTROPHIC MICRO-ORGANISMS
C PREFL(K,M)              PREFERENCE FACTOR FOR CONSUMPTION OF THE M'TH TYPE
C                         OF SUSPENDED DETRITUS BY THE K'TH ANIMAL SPECIES GROUP
C PREFM(K,T)              PREFERENCE FACTOR FOR CONSUMPTION OF THE T'TH TYPE OF
C                         HETEROTROPHIC MICROORGANISMS BY THE K'TH ANIMAL
C                         SPECIES GROUP
C PREFO(K,M)              PREFERENCE FACTOR FOR CONSUMPTION OF THE M'TH TYPE
C                         OF BOTTOM DETRITUS BY THE K'TH ANIMAL SPECIES GROUP
C PREFV(K,I)              PREFERENCE FACTOR FOR CONSUMPTION OF THE I'TH PLANT
C                         SPECIES GROUP BY THE K'TH ANIMAL SPECIES GROUP
C RCONST(M)               THE MAXIMUM RATE OVIPOSITION (PROPORTION OF ADULT
C                         BIOMASS PER TIME UNIT) FOR THE M'TH ANIMAL COHORT
C REPROD(L)               THE PROPORTION OF ADULT BIOMASS TRANSFERRED TO EGGS
C                         PER TIME UNIT IN THE L'TH AINMAL COHORT
C RESP                    RATE OF RESPIRATION PER UNIT BIOMASS PER TIME UNIT
C                         FOR THE ANIMAL COHORT CURRENTLY UNDER CONSIDERATION
C SHELLS                  PROPORTION OF BIOMASS TRANSFERRED TO DETRITUS AS
C                         CAST SKINS DURING A TRANSFER BETWEEN ANIMAL COHORTS
C SHELP                   PROPORTION OF BIOMASS IN A TRANSFER BETWEEN ANIMAL
C                         COHORTS WHICH CONSISTS OF CAST SKINS
C SIEVE(L,N)              RATE OF LOSS OF THE N'TH CHEMICAL CONSTITUENT FROM
C                         THE L'TH ANIMAL COHORT BY LEACHING OR EXUDATION
C SINFUN                  SINE FUNCTION RELATING RATE OF OVIPOSITION TO SEASON
C SLOPE                   RATE OF INCREASE OF LOG (RESPIRATION RATE) WITH LOG
C                         (INDIVIDUAL BIOMASS)
C SUBST(I,N)              THE AMOUNT OF THE N'TH CONSTITUENT OF THE I'TH
C                         SUBSTRATE AVAILABLE TO HETEROTROPHIC MICRO ORGANISMS.
C SUM                     WEIGHTED TOTAL OF FOODS AVAILABLE TO THE ANIMAL
C                         SPECIES GROUP CURRENTLY UNDER CONSIDERATION
C TAKE(K)                 MAXIMUM INTAKE OF CARBON PER UNIT BIOMASS CARBON
C                         PER TIME UNIT IN THE K'TH ANIMAL SPECIES GROUP,
C                         WHEN FOOD SUPPLY IS UNLIMITED
C TAKING                  TOTAL FOOD INTAKE BY THE ANIMAL SPECIES GROUP
C                         CURRENTLY UNDER CONSIDERATION
C THRESH(L)               THE LOWER THRESHOLD VALUE OF TEMPERATURE OR
C                         ACCUMULATED TEMPERATURES FOR TRANSFER INTO THE L'TH
C                         ANIMAL COHORT
C TOT                     SUM OF FOOD PREFERENCE WEIGHTS
C UPTHRE(L)               UPPER THRESHOLD VALUE OF TEMPERATURE FOR TRANSFER
C                         INTO THE L'TH ANIMAL COHORT
C TEMCON(L)               COEFFICIENT FOR TEMPERATURE DEPENDENCE OF TRANSFER
C                         INTO THE L'TH ANIMAL COHORT
      DIMENSION NDRIFA(98),FEEDV(20),FEEDA(98),FEEDL(6),FEEDO(6),           0040
```

```
      1FOOD(98),IAFATE(98),NTRANS(98),SLOPE(98),FEEDM(6),SIEVE(98,6)          0060
      2, ISFATE(98)
      3 ,ATE(5),SUBST(10,6 ),SUBSTQ(10,6 )
C----------------------------------------------------------------------------
C       COMMON BLOCK /ACC/ CONTAINS ACCUMULATED CHANGES, WHICH MAY BE
C       NEGATIVE.  COMMON BLOCK /ACCINC/ CONTAINS THE INCREMENTS TO THE
C       ARRAYS IN /ACC/ FOR A SINGLE TIME UNIT.
C----------------------------------------------------------------------------
        COMMON /ACC/ AGAIN(3,5),EROD(3),H2O(3)                                0100
        COMMON /ACCINC/ AGAINQ(3,5),ERODQQ(3), H2OQQQ(3)                      0120
C----------------------------------------------------------------------------
C       COMMON BLOCK /SPEC/ CONTAINS SPECIFICATIONS AND OTHER INFORMATION
C       COMMON TO THE WHOLE SET OF PROGRAMS, BUT EXCLUDING STATE AND
C       EXOGENOUS VARIABLES.
C----------------------------------------------------------------------------
        COMMON/SPEC/TCOVER,NCHAN,INSTPU(20),    WATER,NSPECV,NSPECA,NORGAN,    0140
      1 NFRACT,NELEM,NOLIT ,NCHECK,IDAY, ATOT, ATOTO,IYRDAY,NREPET(20)        0160
      2,NCOH(20),LISCOH(98),NCOHCU(20),NCOHOR,NFRELM,NFRAC1,NSPCOH,NDEBUG     0180
      3 ,FLOUT,MICROD, PICMIN(98),MONTH
C----------------------------------------------------------------------------
C       COMMON BLOCK /METEOR/ CONTAINS THE VALUES OF EXOGENOUS VARIABLES
C       FOR THE CURRENT TIME UNIT.
C----------------------------------------------------------------------------
        COMMON/METEOR/WIRRIG,ERO,RUNSOL(6),RUNDER(3,6), DARAIN,DAYRUN,
      1EVAP,WATTEM,DAPHOT, DAYRAD, DADUST(3,6) , EXOG(98), RAINCO(6),
      2COMPIN(6), DETIN(5,6), RUNON,FLOWIN,DRIFTV(20,6,6), DRIFTA(98,6),
      3DRIFTM(3,6), DRIFRC(98)
        COMMON/STAT/ CVEC(20,6,6),CORC(5,6),POP(98),CBIOM(98,6),AQUA(6),      0220
C----------------------------------------------------------------------------
C       COMMON BLOCK /STAT/ CONTAINS THE STATE VARIABLES, AND /CHANGE/
C       THEIR INCREMENTS OR DECREMENTS FOR THE CURRENT TIME UNIT.
C----------------------------------------------------------------------------
      1 CLIT(5,6),CBACT(3,6),          DUMMY(96)
        COMMON /CHANCE/ CVECQQ(20,6,6),CORCQQ(5,6),POPQQQ(98),CBIOMQ(98,6)    02E0
      1,AQUAQQ(6), CLITQQ(5,6),CBACTQ(3,6)                ,DUMMQ(96)
C----------------------------------------------------------------------------
C       COMMON BLOCK/PARAM/ CONTAINS THE PARAMETERS OF THE SUBROUTINE.
C----------------------------------------------------------------------------
        COMMON/PARAM/DUMMY1(787),PREFV(98,20),PREFA(98,98),EGCOMP(98,6),      0300
      1PREFL(98,6),PREFO(98,5),PREFM(98,3),TAKE(98),ASSIM(98),                0320
      2AMORTA(98),CURVE(98),THRESH(98),ACCUM(98),AMAX(98),HATCON(98),         03
      3TEMCON(98),HATCOB(98), THRES2(98),RCONST(98),CONSTA(98),CONSTC(98)     03E0
      4,CONSTB(98),CONSA(98),CONSB(98),BOTHRE(98),UPTHRE(98),REPROD(98),      0380
      5SHELF(98),CONCAR(98),CONDAR(98),CONPAR(98),CONST(98),EXOGEN(98),       0400
      6CONTA,CONTB,CONTC,CONS(3),AMIG1(98),AMIG2(98),AMIG3(98)
      7,CONDAR(98),PREFER(5,9),QONE(98), DUMMY2(4)
C----------------------------------------------------------------------------
C       THE NAME LIST CONTAINS THE PARAMETERS THAT ARE TO BE
C       READ IN AT EXECUTION TIME.
C----------------------------------------------------------------------------
        NAMELIST/APUT/ PREFV,PREFA,PREFL,PREFO,PREFM,TAKE,DEPTH,
      *NDPIFA,NTRANS,ASSIM,AMORTA,IAFATE, CURVE,THRESH        ,AMAX,
      *HATCON,TEMCON,HATCOB,THRES2,QONE,
      *                 EGCOMP,RCONST,CONSTA,CONSTC,CONSTB,CONSA,CONSB,
      *BOTHRE,UPTHRE,REPROD,SHELP,CONDAR,CONCAR,CONPAR,CONPAR,
      *STEVE,SLOPE,CONST,EXOGEN,CONTA,CONTB,CONTC,CONS,AMIG1,AMIG2,
      * AMIG3,PREFER,ACCUM,ISFATE
```

```
C-----------------------------------------------------------------------
C     THE FOLLOWING SECTION DEALS WITH ANIMAL CONSUMPTION.
C-----------------------------------------------------------------------

C..........THE AMOUNT OF FOOD AVAILABLE TO AN ANIMAL COHORT IS
C..........DETERMINED BY THE AVAILABILITY OF AND PREFERANCE FOR FOODS
      DO 340 I=1,NSPCOH                                           1280
      FOOD(I) = 0
      SUM=0.                                                      1340
      TOT=0.                                                      1360
      TAKING=TAKE(I)*CBIOM(I,1)                                   1380
      IF (TAKING.LE.0.) GO TO 340                                 1400

C..........OF VEGETATION
      DO 20 J=1,NSPECV                                            1420
      A=PREFV(I,J)                                                1440
      FEEDV(J)=CVEG(J,1,1)*A                                      1460
      IF (FEEDV(J).LE.0.) GO TO 20                                1480
      SUM=SUM+FEEDV(J)                                            1500
      TOT=TOT+A                                                   1520
   20 CONTINUE                                                    1540

C..........OF ANIMAL COHORTS.
      DO 40 J=1,NSPCOH                                            1560
      A=PREFA(I,J)                                                1580
      FEEDA(J)=CBIOM(J,1)*A                                       1600
      IF (FEEDA(J).LE.0.) GO TO 40                                1620
      SUM=SUM+FEEDA(J)                                            1640
      TOT=TOT+A                                                   1660
   40 CONTINUE                                                    1680

C..........OF SUSPENDED AND BENTHIC DETRITUS
      DO 80 J=1,NOLIT                                             1700
      A=PREFL(I,J)                                                1720
      B=PREFO(I,J)                                                1740
      FEEDO(J)=CORG(J,1)*B                                        1760
      IF (FEEDO(J).LE.0.) GO TO 60                                1780
      SUM=SUM+FEEDO(J)                                            1800
      TOT=TOT+B                                                   1820
   60 FEEDL(J)=CLIT(J,1)*A                                        1840
      IF (FEEDL(J).LE.0.) GO TO 80                                1860
      SUM=SUM+FEEDL(J)                                            1880
      TOT=TOT+A                                                   1900
   80 CONTINUE                                                    1920

C..........OF HETEROTROPHIC MICRO ORGANISMS
      DO 100 J=1,MICROB                                           1940
      A=PREFM(I,J)                                                1960
      FEEDM(J)=CBACT(J,1)*A                                       1980
      IF (FEEDM(J).LE.0.) GO TO 100                               2000
      SUM=SUM+FEEDM(J)                                            2020
      TOT=TOT+A                                                   2040
  100 CONTINUE                                                    2060

C..........THE AMOUNT OF FOOD ACTUALLY CONSUMED IS CALCULATED
C..........AND PERCENTAGES REMOVED FROM EACH SOURCE.
      IF (SUM.LE.0.) GO TO 340                                    2080
```

```
      FOOD(I)=SUM/TOT
      FOOD(I)=TAKE(I)*(1.-EXP(-CURVE(I)*FOOD(I)))
      TAKING=FOOD(I)*CBIOM(I,1)

C..........FROM VEGETATION.
      DO 140 J=1,NSPECV
      IF (FEEDV(J).LE.0.) GO TO 140
      FACTOR=TAKING*FEEDV(J)/(SUM*CVEG(J,1,1))
      DO 120 K=1,NELEM
      CVEGQQ(J,1,K)=CVEGQQ(J,1,K)-CVEG(J,1,K)*FACTOR
  120 CONTINUE
  140 CONTINUE

C..........FROM ANIMAL COHORTS.
      DO 180 J=1,NSPCOH
      IF (FEEDA(J).LE.0.) GO TO 180
      FACTOR=TAKING*FEEDA(J)/(SUM*CBIOM(J,1))
      DO 160 K=1,NELEM
      CBIOMQ(J,K)=CBIOMQ(J,K)-CBIOM(J,K)*FACTOR
  160 CONTINUE
  180 CONTINUE

C..........FROM SUSPENDED DETRITUS
      DO 280 J=1,NOLIT
      IF (FEEDL(J).LE.0.) GO TO 220
      FACTOR=TAKING*FEEDL(J)/(SUM*CLIT(J,1))
      DO 200 K=1,NELEM
      CLITQQ(J,K)=CLITQQ(J,K)-CLIT(J,K)*FACTOR
  200 CONTINUE
  220 CONTINUE

C..........FROM BENTHIC DETRITUS.
  240 IF (FEEDO(J).LE.0.) GO TO 280
      FACTOR=TAKING*FEEDO(J)/(SUM*CORG(J,1))
      DO 260 K=1,NELEM
      CORGQQ(J,K)=CORGQQ(J,K)-CORG(J,K)*FACTOR
  260 CONTINUE
  280 CONTINUE

C..........FROM HETERTROPHIC MICRO ORGANISMS.
      DO 320 J=1,MICROB
      IF (FEEDM(J).LE.0.) GO TO 320
      FACTOR=TAKING*FEEDM(J)/(SUM*CBACT(J,1))
      DO 300 K=1,NELEM
      CBACTQ(J,K)=CBACTQ(J,K)-CBACTQ(J,K)*FACTOR
  300 CONTINUE
  320 CONTINUE
  340 CONTINUE
C-----------------------------------------------------------------------
C     RESPIRATION, EXCRETION AND ASSIMILATION IS CALCULATED.
C-----------------------------------------------------------------------

      DO 420 I=1,NSPCOH
      IF (CBIOM(I,1).LE.0.) GO TO 420
      BIOMAS = CBIOM(I,1)/POP(I)
      RESP = EXP(SLOPE(I)* ALOG(CBIOM(I,1)/POP(I)))*POP(I)
     1*CONST(I)*2.**(WATTEM/10.)
```

```
      ASSI=ASSIM(I)*FOOD(I)                                             3020
      EXCR=FOOD(I)-ASSI                                                 3040
      ASSI=ASSI-RESP                                                    3060
      DO 400 K=1,NELEM                                                  3080

C..........THE COHORT GROWS DEPENDENT  ON THE AMOUNT OF FOOD
C..........ASSIMILATED MINUS RESPIRATION.
      CBIOMQ(I,K)=CBIOMQ(I,K)+CBIOM(I,K)*ASSI                           3100
      A=CBIOM(I,K)*EXCR                                                 3120
      CLITQQ(1,K)=CLITQQ(1,K)+A                                         3160

C..........EXCRETA IS ADDED TO THE SUSPENDED DETRITUS WHILE CARBON
C..........FROM RESPIRATION IS ADDED TO THE WATER AS DISSOLVED
C..........INORGANIC CARBON.
      IF (K.GT.1) GO TO 380                                             3200
      AQUAQQ(NFRAC1) = AQUAQQ(NFRAC1)+  RESP*CBIOM(I,K)
      GO TO 400                                                         3240
  380 A=RESP*CBIOM(I,K)                                                 3260
      AQUAQQ(K)=AQUAQQ(K)+A                                             3300
  400 CONTINUE                                                          3320
      BIOMIN(I) = BIOMIN(I)* (1.+ASSI)
  420 CONTINUE                                                          3380
C-----------------------------------------------------------------------
C     THE FOLLOWING SECTION DEALS WITH HETEROTROPHIC MICRO-ORGANISMS.
C-----------------------------------------------------------------------

C..........THE AMOUNT OF FOOD AVAILABLE TO THE MICROBS DEPENDS ON
C..........THE AVAILABILITY OF AND PREFERENCE FOR FOOD.  THE
C..........MICROBS MAY CONSUME SUSPENDED AND BENTHIC DETRITUS
C..........AS WELL AS DISSOLVED SUBSTANCES.
      NOLIT1 = NOLIT+ 1
      NELEM1 = NELEM- 1
      NBACET = 1 + (2*NOLIT)
      DO 430 I=1 , NBACET
      DO 430 J=1 , NELEM
  430 SUBSTQ(I,J) =0.
      DO 440 I=1, NELEM
  440 SUBST(1,I) = AQUA(I)
      DO 460 J=2, NOLIT1
      JJ = J + NOLIT
      DO 460 I=1 , NELEM
      SUBST(J,I) = CLIT(J-1,I)
  460  SUBST(JJ,I) = CORG (J-1,I)
      DO 570 I=1, MICROB
      IF  (CBACT(I,1) .LE.0.) GO TO 570
      SUM =0

C..........THE AMOUNT OF FOOD ACTUALLY CONSUMED IS CALCULATED.
      DO 560 J=1, NBACET
      IF ((SUBST(J,1).LE.0) .OR.(PREFER(I,J).LE.0)) GO TO 560
      SUM = SUM+ (PREFER(I,J) * SUBST(J,1))
  560 ATE(I) = ATE(I) + PREFER(I,J)
      IF ( SUM .LE.0) GO TO 570
      A = SUM / ATE(I)
      B = EXP( CONTA+ CONTB * WATTEM - CONTC* WATTEM*WATTEM)
      FOOD(I) = B/(1.+CONS(I) /A)
  570 CONTINUE
```

```
      DO 600 J=1 , NBACET
      IF ( SUBST(J,1) .LE.O) GO TO 600
      A=1.
      DO 590 K=1, NELEM1
      WANT =0
      DO 580 I=1, MICROB
      IF  ((CBACT(I,1) .LE.0).OR.( ATE(I).LE.O))GO TO 580
      WANT = WANT+(CBACT(I,K)/CBACT(I,1))*(FOOD(I)*PREFER(I,J)/ATE(I))
  580 CONTINUE
      IF (WANT.LE.0) GO TO 600
      A = AMIN1 (A,SUBST(J,K)/WANT)
  590 CONTINUE
      DO 595 K=1, NELEM
      DO 595 I=1, MICROB
      IF (CBACT(I,1) .LE.0) GO TO 595
      C = FOOD(I) * A * CBACT(I,K)/ CBACT(I ,1)
      CBACTQ(I,K) = CBACTQ(I,K)+C
      SUBSTQ(J,K) = SUBSTQ(J,K) - C
  595 CONTINUE
  600 CONTINUE
      DO 620 I=1, NELEM
  620 AQUAQQ(I) = AQUAQQ(I) + SUBSTQ(1,I)
      DO 640 J=2, NOLIT1
      JJ = J + NOLIT
      DO 640 I=1,NELEM
      CLITQQ(J-1,I) = CLITQQ(J-1,I) + SUBSTQ(J,I)
  640 CORGQQ(J-1,I) = CORGQQ(J-1,I) + SUBSTQ(JJ,I)
C-----------------------------------------------------------------------
C     THE FOLLOWING SECTION DEALS WITH LEACHING AD NATURAL MORTALITY.
C-----------------------------------------------------------------------
      DO 940 I=1,NSPCOH                                                 4780
      IF (CBIOM(I,1).LE.0.) GO TO 940                                   4800
      DO 860 K=1,NELEM                                                  4820
      S=CBIOM(I,K)*SIEVE(I,K)                                           4840
      CBIOMQ(I,K)=CBIOMQ(I,K)-S                                         4860
      AQUAQQ(K)=AQUAQQ(K)+S                                             4900
  860 CBIOMQ(I,K)=CBIOMQ(I,K)-AMORTA(I)*CBIOM(I,K)                      4940
      POPQQQ(I) = POPQQQ(I)-AMORTA(I)*POP(I)
      K10=IAFATE(I)
      GO TO (862,864,866,868),K10
  852  CORGQQ(2,K) = CORGQQ(2,K) + AMORTA(I) * CBIOM(I,K)
      GO TO 900
  864  CORGQQ(1,K) = CORGQQ(1,K) + AMORTA(I) * CBIOM(I,K)
      GO TO 900
  866 CLITQQ(2,K)=CLITQQ(2,K)+AMORTA(I)*CBIOM(I,K)
      GO TO 900
  868  CLITQQ(1,K) = CLITQQ(1,K) +AMORTA(I)*CBIOM(I,K)
  900 CONTINUE                                                          5060
C-----------------------------------------------------------------------
C    THE FOLLOWING SECTION DEALS WITH MIGRATION.
C-----------------------------------------------------------------------
      AMIGRA = AMIG3(I) + AMIG1(I) * FOOD(I)
      IF (POP(I).LT.0.) AMIGRA = AMIGRA-AMIG2(I) *POP(I)
      IF (AMIGRA.GE.0) GO TO 940
      POPQQQ(I)=POPQQQ(I) + AMIGRA * POP(I)                             5200
      DO 920 K=1,NELEM                                                  5220
      A=AMIGRA * CBIOM(I,K)                                             5240
```

```
      CBIOMQ(I,K)=CBIOMQ(I,K)+A                                         5260
  920 AGAINQ(2,K)=AGAINQ(2,K)+A                                         5280
  940 CONTINUE                                                          5300
C                                                                       5320
C                                                                       5340
      IF (KDAY.EQ.IDAY) GO TO 980                                       5360
      KDAY=IDAY                                                         5380
      DO 960 I=1,NSPCOH                                                 5400
  960 ACCUM(I)=ACCUM(I)+WATTEM                                          5420
C-----------------------------------------------------------------------
C     THE FOLLOWING SECTION ALLOWS FOR TRANSFER FROM ONE COHORT TO
C     ANOTHER.
C-----------------------------------------------------------------------

C..........J4 GIVES THE ADDRESS IN THE TRANSFER ARRAY.
C..........J GIVES THE COHORT NUMBER WITHIN THE SPECIES.
  980 J4=0                                                              5560
      FACTOR = 1. - EXP(-FLOUT*.00001/DEPTH)
      DO 1460 I=1,NSPECA                                                5580
      J3=NCOH(I)                                                        5600
      J5 = J4 + J3
      DO 1440 J=1,J3                                                    5620
      J4=J4+1                                                           5640
      K1=NTRANS(J4)                                                     5700
      IF (J.LE.1) GO TO 1140                                            5660
      IF (CBIOM(J4-1,1).LE.0.) GO TO 1320                               5680
      GO TO (1000,1020,1000,1040),K1                                    5720
 1000 IF (ACCUM(J4).LE.THRESH(J4)) GO TO 1320                           5740
      HATCH=(ACCUM(J4)-THRESH(J4))/(AMAX(J4)-THRESH(J4))                5760
      IF (ACCUM(J4).GE.AMAX(J4)) ACCUM(J4)=0.                           5780
      GO TO 1060                                                        5800
 1020 HATCH=AMIN1(1.,AMAX1(HATCON(J4)*EXP(TEMCON(J4)*WATTEM)-HATCOB(J4) 5820
     1,0.))                                                             5840
      GO TO 1060                                                        5860
 1040 CONTINUE                                                          5880
 1060 IF(POP(J4-I).LE.0.) GO TO 1320
      CHANGE=HATCH*POP(J4-1)                                            5900
      POPQQQ(J4-1)=POPQQQ(J4-1)-CHANGE                                  5920
      IF (K1.NE.3) POPQQQ(J4)=POPQQQ(J4)+CHANGE                         5940
      CHANGE = HATCH*((2.-HATCH)-POP(J4-1)*BIOMIN(J4-1)*
     1 (1.-HATCH)/CBIOM(J4-1,1))
      BIONEX=2.*CBIOM(J4-1,1)*(1.-HATCH)/POP(J4-1)-BIOMIN(J4-1)
     1 *(1.-2.*HATCH)
      BIOMIN (J4) = AMIN1(BIOMIN(J4), BIONEX)
      SHELLS=SHELP(J4-1)*CHANGE                                         6100
      DO 1120 K=1,NELEM                                                 6120
      A= CBIOM(J4-1,K)
      IF(A.LE.0.) GO TO 1120
      K11 = ISFATE(I)
      GO TO (1070,1072,1074,1076),K11
 1070 CORGQQ(2,K)=CORGQQ(2,K)+SHELLS*A
      GO TO 1078
 1072 CORGQQ(1,K)=CORGQQ(1,K)+SHELLS*A
      GO TO 1078
 1074 CLITQQ(2,K)=CLITQQ(2,K)+SHELLS*A
      GO TO 1078
 1076 CLITQQ(1,K)=CLITQQ(1,K)+SHELLS*A
```

```
 1078 CBIOMQ(J4-1,K) = CBIOMQ(J4-1,K) - CHANGE *A
      IF (K1.NE.3) GO TO 1100                                            6180
      AGAINQ(1,K)=AGAINQ(1,K)-(CHANGE-SHELLS) *A                         6200
      GO TO 1120                                                         6220
 1100 CBIOMQ(J4,K)=CBIOMQ(J4,K)+(CHANGE-SHELLS)  *A                      6240
 1120 CONTINUE                                                           6260
      GO TO 1320                                                         6280
C-----------------------------------------------------------------------
C     THE FOLLOWING SECTION DEALS WITH OVIPOSITION.
C-----------------------------------------------------------------------
 1140 IF ((K1.GT.1).AND.(CBIOM(J3,1).LE.0.)) GO TO 1320                  6380
      GO TO  (1160,1200,1220,1260),K1                                    6400
 1160 POPQQQ(J4)=POPQQQ(J4)+EXOGEN(J4)                                   6420
      DO 1180 K=1,NELEM                                                  6440
      A= EXOGEN(J4) * EGCOMP (I,K)
      AGAINQ(1,K) = AGAINQ (1,K) + A
 1180 CBIOMQ (J4,K) = CBIOMQ(J4,K) + A
      GO TO 1320                                                         6480
 1200 IF (WATTEM.LT.THRESH(J4)) GO TO 1320                               6500
      HATCH=REPROD(J4)                                                   6520
      GO TO 1280                                                         6540
 1220 HATCH=0.                                                           6560
      IF (WATTEM.LT.THRESH(J4).OR.WATTEM.GT.UPTHRE(J4)) GO TO 1320       6580
      HATCH=RCONST(J4)                                                   6600
      IF (CONSTC(J4).LE.0.) GO TO 1240                                   6620
      SINFUN=CONSTA(J4)+(CONSTC(J4)*SIN(.0174*(FLOAT(IYRDAY)+CONSTB(J4))  6640
     1))                                                                 6660
      IF (SINFUN.LE.0.) GO TO 1240                                       6680
      HATCH=HATCH*SINFUN                                                 6700
 1240 IF (CONSB(J4).EQ.0.) GO TO 1280                                    6720
      POPFUN=AMIN1(1.,AMAX1(0.,(CONSA(J4)+CONSB(J4)*LOG(POP(J5)))))      6740
      HATCH=HATCH*POPFUN                                                 6760
      GO TO 1280                                                         6780
 1260 CONTINUE                                                           6800
C-----------------------------------------------------------------------
C     THE FOLLOWING SECTION DEALS WITH REPRODUCTION.
C-----------------------------------------------------------------------
 1280 CHANGE=HATCH*CBIOM(J5,1)
      IF (CHANGE.LE.0.) GO TO 1320
      DO 1300 K=1,NELEM                                                  6900
      CHELEM=CHANGE*EGCOMP(I,K)/EGCOMP(I,1)                              6920
      CBIOMQ(J5,K)=CBIOMQ(J5,K)-CHELEM
 1300 CBIOMQ(J4,K)=CBIOMQ(J4,K)+CHELEM                                   6960
      POPQQQ(J4)=POPQQQ(J4)+CHANGE/EGCOMP(I,1)                           6980
C-----------------------------------------------------------------------
C     THE FOLLOWING SECTION DEALS WITH DRIFT.
C-----------------------------------------------------------------------
 1320 IF (POP(J4).LE.0) GO TO 1405
      K1 = NDRIFA(J4)
      GO TO (1440,1340, 1370, 1420), K1                                  7100
 1340 POPQQQ(J4) = POPQQQ(J4) - POP(J4) * FACTOR                         7120
      DO 1360 K = 1, NELEM                                               7140
      A = CBIOM(J4,K) * FACTOR                                           7160
      CBIOMQ(J4,K) = CBIOMQ(J4,K) - A                                    7180
      AGAINQ(2,K) = AGAINQ(2,K) - A                                      7200
 1360 CONTINUE                                                           7220
      GO TO 1440                                                         7240
```

```
 1370 A = ALOG(POP(J4))
      PARAB= CONPAR(J4)+CONBAR(J4)*A+CONCAR(J4)*A*A
      IF(FOOD(J4).GT.0.) GO TO 1380
      PARAB = 1.
      GO TO 1390
 1380 PARAB = AMAX1(0.,AMIN1(1.,PARAB-CONDAR(J4)*ALOG(FOOD(J4))))
      IF (PARAB .LE.0.) GO TO 1405
 1390 P = PARAB * FACTOR
      DO 1400 K = 1, NFLEM
      DROUT = CBIOM(J4,K) * P
      CBIOMQ(J4,K) = CBIOMQ(J4,K) - DROUT
      AGAINQ(2,K) = AGAINQ(2,K) - DROUT
 1400 CONTINUE
      DROUT = POP(J4) * P
      POPQQQ(J4) = POPQQQ(J4) - DROUT
 1405 POPQQQ(J4) = POPQQQ(J4)+DRIFPO(J4)*FLOWIN
      DO 1418 K=1,NFRELM
      A= DRIFTA(J4,K) * FLOWIN
      CBIOMQ(J4,K)=CBIOMQ(J4,K)+A
 1418 AGAINQ(2,K) = AGAINQ(2,K) +A
      GO TO 1440
 1420 CONTINUE
 1440 CONTINUE
 1460 CONTINUE
      DO 1470 I = 1, MICROB
      DO 1470 K = 1, NFRELM
      A = CBACT(I,K)*FACTOR
      CBACTQ(I,K) = CBACTQ(I,K) - A
 1470 AGAINQ(2,K)=AGAINQ(2,K)-A
      IF (NDEBUG.LE.0) RETURN
      WRITE (6,1480)
 1480 FORMAT (////)
      WRITE(6,1485)
C-----------------------------------------------------------------------
C     THE FOLLOWING ALLOWS FOR THE PRINTING OF DECREMENTS TO
C     THE STATE VARIABLES
C-----------------------------------------------------------------------
 1485 FORMAT(25X,'  ANIMAL NDEBUG')
      WRITE(6,1490) ((AGAINQ(I,J), I=1,3), J=1,5)
 1490 FORMAT (' AGAINQ ', 10F12.6)
      WRITE (6,1500)  ((CVEGQQ(I,1,K),I=1,NSPECV),K=1,NFRELM)
 1500 FORMAT (' CVEGQQ   ', 10F12.6)
 1520 FORMAT (' CBIOMQ   ', 10F12.6)
      WRITE (6,1520)((CBIOMQ(I,K), I=1,NSPCOH),K=1,NFRELM)
      WRITE (6,1540) (POPQQQ(I), I=1,NSPCOH)
 1540 FORMAT (' POPQQQ   ', 10F12.6)
      WRITE (6,1560) ((CLITQQ(I,K), I=1,NOLIT),K=1,NFRELM)
 1560 FORMAT (' CLITQQ   ', 10F12.6)
      WRITE (6,1580)  ((CORGQQ(I,K), I=1,NOLIT), K=1,NFRELM)
 1580 FORMAT (' CORGQQ   ', 10F12.6)
      WRITE (6,1600)  (AQUAQQ(K), K=1,NELEM)
 1600 FORMAT (' AQUAQQ   ', 10F12.6)
      WRITE (6,1620) ((CBACTQ(I,J),I=1,MICROB),J=1,NFRELM)
 1620 FORMAT (' CBACTQ   ',10F12.6)
      WRITE (6,1640) (BIOMIN(I), I=1, NSPCOH)
 1640 FORMAT ( ' BIOMIN ', 10F12.6)
      RETURN
```

```
C-----------------------------------------------------------------------
C     THE FOLLOWING ALLOWS FOR READING OF PARAMETERS.
C-----------------------------------------------------------------------
      ENTRY AINPUT
      READ (5,APUT)
      RETURN
      END
```

PLANT SUBROUTINE

Donald Porcella, Derry Koob, William Grenney and

Joseph Wlosinski

INTRODUCTION

This submodel is designed for use as part of a "package" of programs for modelling aquatic systems (creeks, or spring pools) in arid lands. It is for general use in that many of the characteristics of the system may be specified at execution time, though in its present form this sub-model imposes a number of restrictions on those specifications. For the general features of this modelling "package", the reader is referred to the introduction to this series of reports.

VERBAL DESCRIPTION

The sub-program calculates all changes in the plant material in the system, with the exception of those due to herbivory, and all the associated changes in the water and detritus. Each plant group is treated as located at a constant mean depth in the water, and, if rooted, may be subject to breakage. Emergent vegetation is not provided for, and it is assumed that material of each plant species group may be regarded as uniform, without differentiation into organs.

Photosynthesis depends above all on the radiation reaching the plant. The exogenous variables about which information is supplied to this subroutine include the photoperiod, and the mean radiation intensity at the water surface during the daylight hours. From this, it is necessary for the subroutine to calculate the radiation received by each plant species. Attenuation of the radiation by the water mass and by suspended sediment is assumed dependent on the length of the water column above the plant tissue -- that is, on its depth in the water -- by an exponential decay function as shown in Fig. 1. The effect of sediment of a particular type is proportional to the weight contained in that column of water, inorganic sediment having less effect, weight for weight, than organic sediment. Effects on radiation quality are ignored. Any plant at a shallower depth than that under consideration also, like suspended sediment or detritus has a shading effect, assumed proportional to the carbon content of the plant biomass, self-shading is also provided for, half the biomass of the same species (or any other at the same mean depth) being included in the shading material.

Besides the radiation, photosynthesis is also affected by temperature (an exogenous variable) and by the availability within the plant of nutrients needed to convert photosynthate into new cell materials. The temperature effect is assumed parabolic (Fig. 2). The internal availability of nutrients is expressed as the ratio of the amount of each nutrient in the plant tissue to its carbon content, and it is assumed that the dependence of photosynthesis on this quantity is exponential (Fig. 3). Its dependence on radiation intensity follows the Mitscherlich law (Fig. 4). The rate of photosynthesis realized is thus calculated by multiplying the product of the quantities indicated in the preceding sentences by a constant for the species group representing its maximum possible hourly photosynthesis under optimal conditions (a purely theoretical quantity, for it would be attained only if internal nutrient concentrations were indefinitely large) and by the photoperiod. All constants in these expressions are special to each plant group included as a separate entity in the model. Photosynthesis will take place only to the extent that the requisite material-bicarbonate is available in the surrounding water (Figs. 5 and 6). The rate at which nutrients are taken up to "match" the photosynthesis is proportional to the difference between the internal concentration and that which would be in equilibrium with the existing external concentration; if the external supply is constantly renewed, then the internal concentration will be asymptotic to this equilibrium value.

The demands by the various plant groups for bicarbonate and for mineral nutrients from the surroundings are summed, and compared with the amounts available. If any is inadequate, all demands for that substance are reduced in such proportion as can be met.

Plants growing on the bottom are assumed to be fixed, and cannot be supplemented or lost by drift. Planktonic species on the other hand may enter the system by surface inflow; and if water is lost (otherwise than by evaporation or underground seepage) plankton will be lost too. Part of the rooted angiosperms may become detached, and these then follow the same pattern as the planktonic algae; the proportion which becomes detached in this way each day is assumed to be a constant.

There is some leakage of carbon compounds from the plant to the surrounding water, at a constant proportional rate for each plant species. Net losses of other elements may also occur if the content of the element is greater than th t which is in equilibrium with the surroundings (Fig. 6).

The plants are also subject to mortality -- again at a constant proportional rate for each species group. The dead material has a fate which may differ from group to group, and is specified for each in the NAMELIST information read in the first time this subroutine is called; planktonic species will be transferred to suspended detritus of the appropriate category, benthic species to bottom sediments; floating portions of rooted species are treated as planktonic, the rest as benthic.

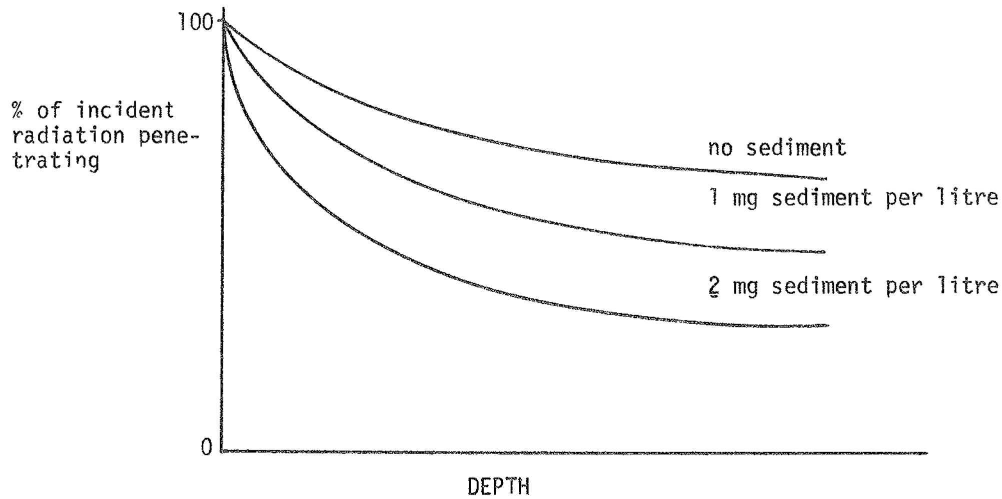


Figure 1. Relationship between depth of the water column and radiation intensity reaching submerged plants.

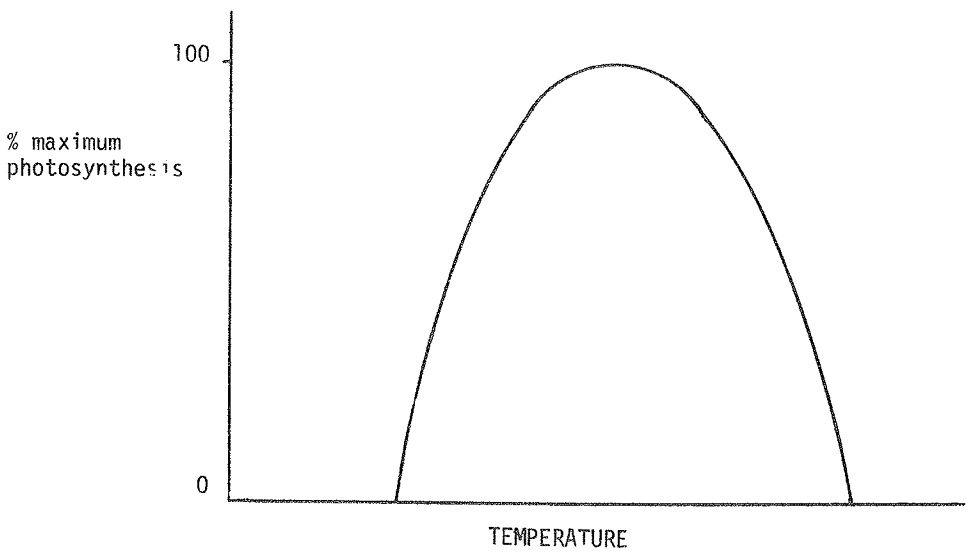


Figure 2. Relationship between water temperature and photosynthetic rate.

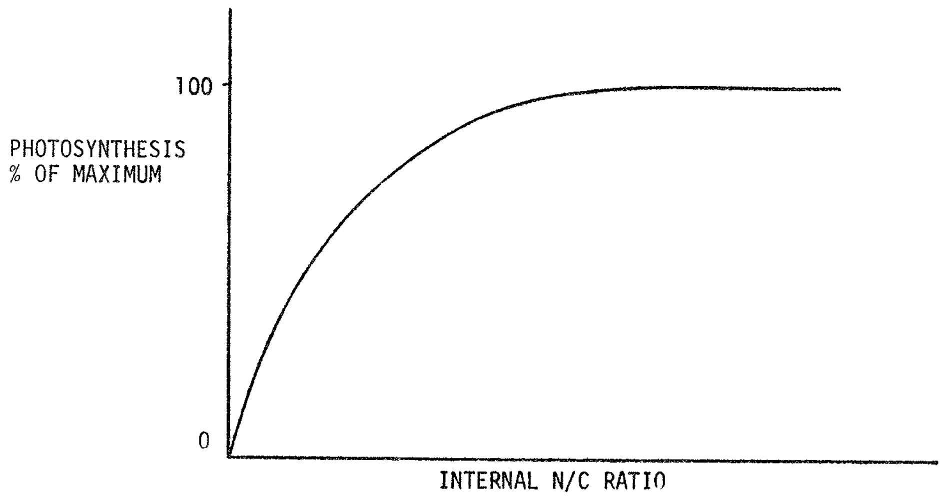


Figure 3. Photosynthetic rate as affected by plant nutrient status

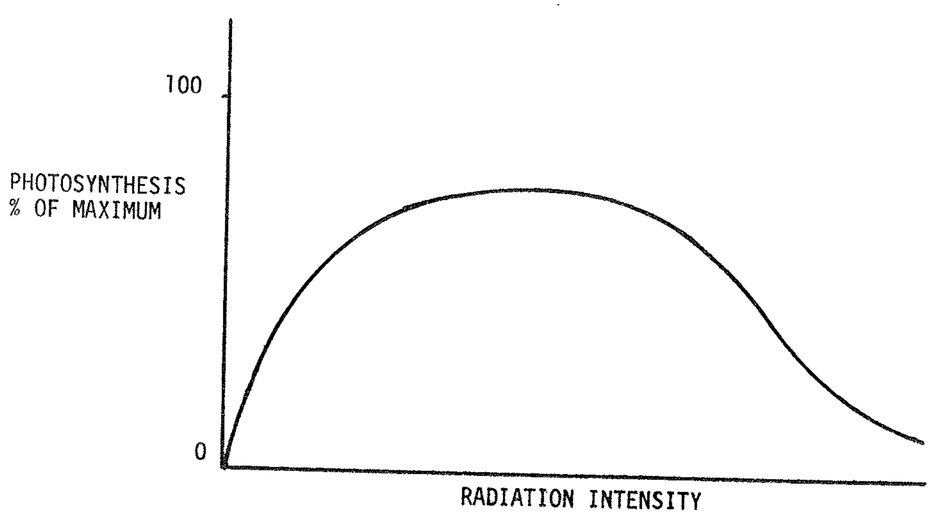


Figure 4. Photosynthesis rate as a function of radiation intensity

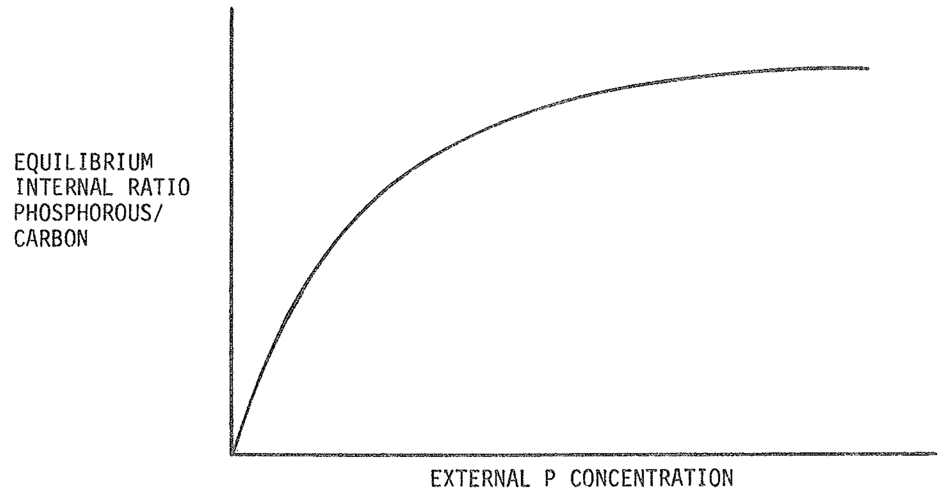


Figure 5. Change in equilibrium of internal phosphorus/carbon ratio as P concentration increases in the medium.

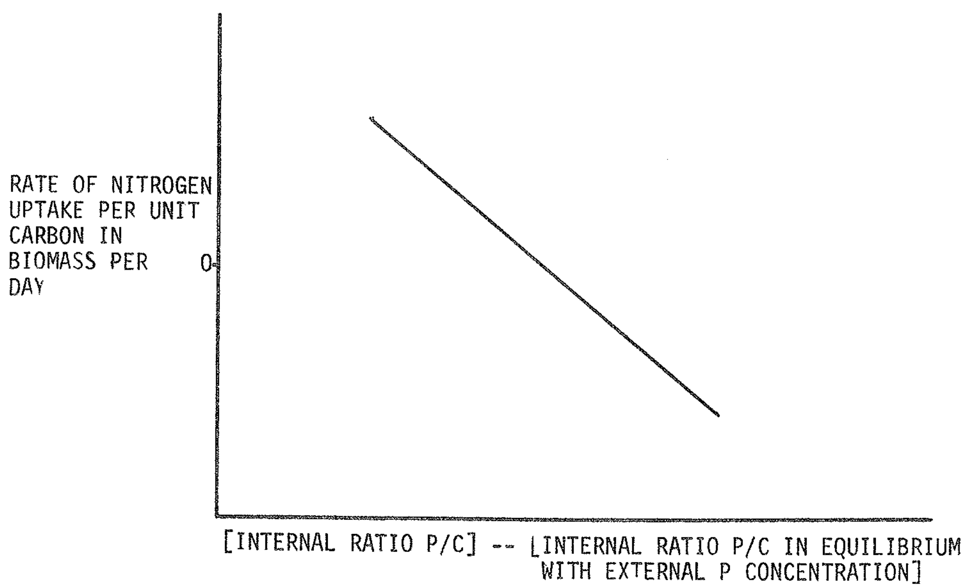


Figure 6. Rate of nitrogen uptake in relation to phosphorus/carbon ratios in the tissue and the medium.

Assumptions

Some of the leading assumptions involved in this sub-model are as follows:

1. The absorption of radiation by suspended inorganic material is proportional to the weight of all constituents, that by organic material is proportional to its carbon content. All categories of suspended plant material absorb radiation to the same extent. Absorption of radiation by zooplankton can be ignored.
2. Photosynthesis rate is proportional to the photoperiod and to a factor depending on mean radiation intensity; the distribution of radiation intensity through the daylight hours can be ignored.
3. Dependence of photosynthesis on temperature and mean radiation intensity can be expressed by optimum-type curves of fixed form.
4. Photosynthesis may be limited by the internal supply of other elements required for biosyntheses, the potential rate increasing to an asymptote as the internal ratio of the other element to carbon increases.
5. For any given external concentration of a nutrient element, there is an equilibrium internal concentration, and the rate of uptake or loss of each element is proportional to the difference between the internal concentration and this equilibrium concentration.
6. Plant tissue dies at a rate constant for each species group, and is transferred to a detritus category constant for that group.

Mathematical Description

Photosynthesis and growth by the various groups of plant species are first calculated on the assumption that the raw materials in the system are adequate to cover all needs. The total requirements are then compared with supplies, and if necessary the actual photosynthesis and growth are scaled down from these potential figures.

The light intensity Z_{1_p} at a depth P_{1_p} below surface, available for photosynthesis by plant species group p, is

$$Z_{1_p} = V_1 \cdot \exp(P_2 \cdot P_{1_p} + P_3 \cdot Z_2 + P_4 \cdot Z_3 + P_5 \cdot Z_4) \qquad (1)$$

where V_1 is the incident radiation intensity, and P_2, P_3, P_4, P_5 are (negative) extinction coefficients appropriate respectively to the water mass (including solutes and colloidal material); inorganic solids in suspension at shallower depths; organic detritus suspended at shallower depths; and plant material floating at shallower depths. The quantities Z_2, Z_3, and Z_4 are the quantities of material of each type intercepting light, defined as:

where $\frac{1}{2} X_{21_{3c}}$ is proportional to the weight of inorganic solids in suspension per unit area, and U_1 is the mean depth of water,

$$Z_3 = (X_{21_{11}} + X_{21_{21}}) \cdot P_{1_p} / U_1 \tag{3}$$

where $(X_{21_{11}} + X_{21_{21}})$ is the weight of suspended detritus (in terms of carbon); and

$$Z_4 = \sum_{p \in S} X_{1_{p1}} + \frac{1}{2} \sum_{p \in T} X_{1_{p1}} \tag{4}$$

where $X_{1_{i1}}$ is the weight of carbon in the i'th plant species group, and:

$$S = \{i\} \quad , \quad P_{1_i} < P_{1_p}$$

$$T = \{i\} \quad , \quad P_{1_i} = P_{1_p}$$

That is, S is the set of those plant species groups at a shallower mean depth than p, and T is the set of those at the same mean depth.

The rate of photosynthesis per hour, at optimum temperature and internal nutrient concentration, is then:

$$Z_{5_p} = (Z_{1_p} / P_{6_p}) \cdot \exp(1 - Z_{1_p} / P_{6_p}) \tag{5}$$

where P_{6_p} is the radiation intensity optimal for the p'th plant species. The effect of temperature is to multiply this optimum rate by a factor:

$$Z_{6_p} = P_{7_p} + P_{8_p} \cdot V_2 + P_{9_p} \cdot V_2^2 \tag{6}$$

where V_2 is the water temperature and P_{7_p}; P_{8_p}; P_{9_p}; are constants for the species. The nutrient content of the tissues also affects the rate. It is assumed that this effect follows a Mitscherlich law, and that the nutrient content can be expressed as the ratio of nutrient to carbon in the plant. Thus a further multiplying factor is required

$$Z_{7_p} = \prod_{c \in M} \{1 - \exp(P_{10_{pc}} - P_{11_{pc}} X_{1_{pc}} / X_{1_{p1}})\} \tag{7}$$

where $M = \{2, 3, 4\}$
is the set of nutrients affecting photosynthesis, and P_{10_p}, P_{11_p} are constants.

The potential rate of photosynthesis per hour, by the p'th plant species, per unit weight of carbon in photosynthesizing organs, is accordingly:

$$Z_8 = Z_5 \cdot Z_6 \cdot Z_7 \cdot X_1 \tag{8}$$

The respiration rate of the p'th plant species group per day is:

$$_R\dot{X}_{1_{p1}} = P_{1_{1p}} \cdot \exp\,(P_{13_p} \cdot V_2) \cdot X_{1_{p1}} \tag{9}$$

where P_{12_p}, P_{13_p} are constants.

The requirement for carbon to meet the potential net photosynthesis rate of the p'th plant species group is thus:

$$Z_{9_{p1}} = X_{1_{p1}} (V_3 \cdot Z_{8_p} - {_R\dot{X}_{1_{p1}}}) \tag{10}$$

where V_3 is the daily photoperiod in hours.

The bicarbonate available to supply the carbon for photosynthesis is then tested. If this, together with the total respiratory carbon from heterotrophic organisms, is inadequate, photosynthesis is scaled down proportionally. Thus

$$_P\dot{X}_{1_{p1}} = \min\,\{\sum_p Z_{9_{p1}}, (X_{24_b} + Z_{10})\} \cdot Z_{9_{p1}} / \sum_p Z_{9_{p1}} \tag{11}$$

where $_P\dot{X}_{1_{p1}}$ is the flux of carbon into the p'th plant species group due to photosynthesis, X_{24_b} is the content of bicarbonate carbon in the water, and Z_{10} is the total respiratory carbon per time unit (day) from heterotrophic organisms.

The rate of uptake of other nutrients from the water is proportional to the differences between the existing internal nutrient concentration and a value which would be in equilibrium with the external concentration. This equilibrium concentration will be

$$Z_{11_{pc}} = P_{14_{pc}} \cdot \{1 - \exp(P_{15_{pc}} \cdot Z_{11_c})\} \tag{12}$$

where Z_{11_c}, is the external concentration of the c'th element, is

$$Z_{12_c} = X_{24_c} \cdot 10^{-6}/U_1 \tag{13}$$

and P_{14_c}, P_{15_c} are constants.

The potential rate of uptake will be

$$Z_{13_{pc}} = P_{16_{pc}} X_{1_{p1}} \{Z_{12_{pc}} - X_{1_{pc}} / X_{1_{p1}}\} \tag{14}$$

This daily rate may be negative if the external concentration is low, but cannot exceed the content of nutrient in the plant tissue. Daily uptake by all plants together cannot exceed the content in the water, and if the potential uptake

$$Z_{14_{pc}} = 1 \qquad , \sum_p Z_{13_{pc}} \leq X_{24_c}$$

$$Z_{14_{pc}} = X_{24_c} / \sum_p Z_{13_{pc}} \quad \sum_p Z_{13_{pc}} > X_{24_c}$$

where X_{24_c} is the quantity of the c'th material in solution, then

$$_U\dot{X}_{1_{pc}} = \max\,(Z_{14_{pc}} \cdot Z_{13_{pc}}, -X_{1_{pc}}) \quad , \qquad c\varepsilon M \tag{15}$$

where $_U\dot{X}_{1_{pc}}$ is the uptake (or loss) of the c'th element by the p'th plant species group.

The corresponding energy increment is proportional to the increment in carbon, thus.

$$_P\dot{X}_{1_{p5}} = X_{1_{p1}} \cdot X_{1_{p5}} / X_{1_{p1}} \tag{16}$$

The pH and bicarbonate concentration are assumed to be such that the system is well buffered. Thus, over the time step, carbon is assumed to be exchanged with the atmosphere at the same rate as it is used in photosynthesis. Net photosynthesis constitutes a gain to the ecosystem as a whole. Energy is also a net gain. We express this as.

$$_P\dot{X}_{0_{11}} = -\sum_p {_P}X_{1_{p1}} \tag{17}$$

$$_P\dot{X}_{0_{15}} = -\sum_p {_P}X_{1_{p5}} \tag{18}$$

where $X_{0_{rc}}$ indicates the rate of gain or loss of the c'th component to the ecosystem as a whole through the r'th channel (1 being the atmosphere or space), and $_P\dot{X}_{0_{rc}}$ indicates that part of the flux ascribable to photosynthesis.

Mineral nutrients required for growth are derived from ions in solution, thus,

$$_U\dot{X}_{24_c} = -\sum_p {_P}\dot{X}_{1_{pc}} \quad , \qquad c\varepsilon M \tag{19}$$

For each plant species group, a proportion of the tissue P_{17_p} may be able to leave the system, moving freely with and in proportion to the outflow water. For phytoplankton, P_{17_p} is unity; for benthic species it may be zero; in some aquatic angiosperms, portions tend to break off, and here the value of P_{17_p} will be intermediate. In any case the net change by drift in the c'th constituent of the p'th plant species group is:

$$_{D}\dot{X}_{1_{pc}} = V_{4_{pc}} P_{17_p} X_{1_{pc}} \; _{F}\dot{X}_{25} \cdot 10^{-6}/U_1 \tag{20}$$

where $\dot{X}_{25}$ is the rate of flow of water out of the system and $V_{4_{pc}}$ is the rate of drift of the same type of material into the system.

The plant tissues are liable to lose organic solutes to the surrounding water at a proportional rate $P_{15_{pc}}$ specific to the p'th plant species group. This affects carbon only, possible net losses of other constituents are covered by (14) above. Thus,

$$_{L}\dot{X}_{2_{pc}} = P_{18_{pc}} X_{1_{pc}} \tag{21}$$

$$_{L}\dot{X}_{24_c} = -\sum_{p} {_{L}\dot{X}_{1_{pc}}} \tag{22}$$

where $_{L}\dot{X}_{1_{pc}}$ is the loss from the plant tissue due to leaching and $_{L}\dot{X}_{24_c}$ the corresponding gain to the water. A constant rate of natural mortality P_{19_p} is also assumed, leading to a transfer of biomass constituents to the bottom sediments in the case of benthic species, to suspended detritus in other groups -- to the fine or coarse fraction, as the case may be:

$$_{M}\dot{X}_{1_{pc}} = -P_{19_p} X_{1_{pc}} \tag{23}$$

$$_{P}\dot{X}_{22_{1c}} = -\sum_{p\epsilon A} {_{M}\dot{X}_{1_{pc}}} \tag{24}$$

$$_{P}\dot{X}_{22_{2c}} = -\sum_{p\epsilon A} {_{M}\dot{X}_{1_{pc}}} \tag{25}$$

$$_{P}\dot{X}_{21_{1c}} = -\sum_{p\epsilon C} {_{M}\dot{X}_{1_{pc}}} \tag{26}$$

$$_{P}\dot{X}_{21_{2c}} = \sum_{p\epsilon D} -{_{M}\dot{X}_{1_{pc}}} \tag{27}$$

Where $_{P}\dot{X}_{22_{1c}}, {_{P}\dot{X}_{22_{2c}}}$ are the rates of addition of plant material to the bottom sediments, fine sand and coarse fractions respectively; $_{P}\dot{X}_{21_{1c}}, {_{P}\dot{X}_{21_{2c}}}$ are the corresponding fluxes to the suspended detritus fractions; and A,B,C,D are the sets of plant species which are small in size and benthic in habit, large benthic, small planktonic and large planktonic respectively ("planktonic" here being taken to include species which, though rooted, may release floating portions).

SUBMODEL LIMITATIONS

For this submodel, some features may be specified at execution time through the calling program, while others are inflexibly determined by the built-in features of the submodel:

Inflexible

1. The carbon content of each component is treated as a whole, and not divided into different groups of chemical components. Carbon occupies first place in the list of constituents tracked through time.
2. The chemical energy contained in the organic or biological components is included in the model, and occupies last place in the list of constituents tracked through time.
3. In its present state the submodel provides only for four groups of plant species, of which the first two are planktonic, and the third benthic, while the fourth consists of rooted macrophytes of which portions may break away. To make these specifications flexible would, however, require only minor changes in the program.
4. No plant organs are distinguished.
5. The submodel assumes only three categories of detritus, suspended and at the bottom, of which the third is inorganic while the first and second are organic material, fine and coarse in size respectively.

Flexible

1. The number of chemical elements contained in the plant material, other than carbon, which are to be modelled may take any value from zero to the limit imposed by the program dimensions.
2. Rates may be calculated for a time step which is any multiple of 24 hours. Shorter time steps are not envisaged. There is, for instance, no provision for calculating separate rates of change during the daytime and nighttime hours.

Limitations imposed by array dimensions are, in the main, less severe than limitations due to program structure. The arrays special to this subroutine, and the features on which they might impose limitations are as follows:

AEXT(a)	PHOTA(c)	SPREQ(c,b)
AINT(a)	PHOTD(c)	
AMORT(c)	PLDEP(c)	UPCON(c,a)
CONNIT(c,a)	PRODRF(c)	UPCON1(c,a)
CONNI2(C,A)		UPCON2(c,a)
CONRAD(c)	REQ(b)	
CONTE1(c)	RESPC(c)	
CONTE2(c)	RESPD(c)	
CONTE3(c)	SIEVEG(c)	

Where	FORTRAN Equivalent
a. Number of chemical constituents (excluding energy)	NELEM1
b. Number of chemical constituents (including energy)	NELEM
c. Number of plant species groups	NSPECV

Table of symbols

Symbol	FORTRAN	Equations
P_{1p}	PLDEP(P)	1
P_{2p}	EXTINW	1
P_{3p}	EXTINS	1
P_{4p}	EXTIND	1
P_{5p}	EXTINP	1
P_{6p}	CONRAD(P)	5
P_{7p}	CONTE1(P)	6
P_{8p}	CONTE2(P)	6
P_{9p}	CONTE3(P)	6
P_{10pc}	CONNI2(P,C)	7
P_{11pc}	CONNIT(P,C)	7
P_{12p}	RESPC(P)	9
P_{13p}	RESPD(P)	9
P_{14pc}	UNCON1(P,C)	12
P_{15pc}	UPCON2(P,C)	12
P_{16pc}	UPCON(P,C)	14
P_{17p}	PRODRF(P)	20
P_{18p}	LEACH(P)	21
P_{19p}	AMORT(P)	23
V_1	DEPTH	2
V_1	DAYRAD	1
V_2	WATTEM	6
V_3	DAPHOT	10
V_{4pc}	DRIFTV(P,1,C)	20

Continued

Table of symbols

Symbols	FORTRAN	Equations
$X_{0_{rc}}$	AGAIN(R,C)	16
${}_P\dot{X}_0$	AGAIN(R,C)[in part]	17
${}_P\dot{X}_0$	AGAIN(R,C)[in part]	18
$\dot{X}_{1_{pc}}$	CVEG(P,1,C)	4
$\dot{X}_{1_{pc}}$	CVEGQQ(P,1,C)	
${}_D\dot{X}_{1_{pc}}$	CVEGQQ(P,1,C)[in part]	20
${}_L\dot{X}_{1_{pc}}$	CVEGQQ(P,1,C)[in part]	21
${}_M\dot{X}_{1_{pc}}$	CVEGQQ(P,1,C)[in part]	23
${}_P\dot{X}_{1_{pl}}$	CVEGQQ(P,1,C)[in part]	11
${}_R\dot{X}_{1_{pl}}$	CVEGQQ(P,1,C)[in part]	9
${}_U\dot{X}_{1_{pc}}$	CVEGQQ(P,1,C)[in part]	15
$\dot{X}_{21_{dc}}$	CLIT(D,C)	2
$\dot{X}_{21_{dc}}$	CLITQQ(D,C)	
${}_P\dot{X}_{21_{dc}}$	CLITQQ(D,C)[in part]	26, 27
$\dot{X}_{22_{dc}}$	CORG(D,C)	
$\dot{X}_{22_{dc}}$	CORGQQ(D,C)	
${}_P\dot{X}_{22_{dc}}$	CORGQQ(D,C)[in part]	24, 25
$\dot{X}_{24_c}$	AQUA(C)	11
$\dot{X}_{24_c}$	AQUAQ(C)	19
${}_L\dot{X}_{24_c}$	AQUAQ(C)[in part]	22
${}_U\dot{X}_{24_c}$	AQUAQ(C)[in part]	19
$\dot{X}_{25}$	FLOUT	20

Continued

Table of symbols

Symbol	FORTRAN *	Equations
Z_{1_p}	RADIA	1
Z_2	SED	2
Z_3	DETRI	3
Z_4	A	4
Z_{5_p}	----	5
Z_{6_p}	----	6
Z_{7_p}	A	7
Z_{8_p}		8
$Z_{9_{pc}}$	ASPREQ(P,C)	10
Z_{10}	----	11
Z_{11_c}	AEXT(C)	12
$Z_{12_{pc}}$	EQUIL	13
$Z_{13_{pc}}$	SPREQ(P,C)	14

*Where the mathematical symbol is a time derivative, the FORTRAN equivalent is an increment over a time step.

```
      SUBROUTINE VEGET                                                  CD2P

C AEXT(N)           THE CONCENTRATION IN WATER (G.PER G.) OF THE N'TH
C                   CHEMICAL CONSTITUENT
C ATNT(N)           THE RATIO IN PLANT TISSUE OF THE N'TH CHEMICAL
C                   CONSTITUENT TO CARBON
C AMORT(I)          THE PROPORTIONAL MORTALITY PER TIME UNIT OF THE
C                   I'TH PLANT SPECIES GROUP
C CONNT1(I,N)       PROPORTIONALITY CONSTANT FOR THE EFFECT OF THE N'TH
C                   CONSTITUENT ON RATE OF PHOTOSYNTHESIS IN THE I'TH
C                   PLANT SPECIES GROUP
C CONNT2(I,N)       LN (PROPORTIONAL REDUCTION IN RATE OF PHOTOSYNTHESIS)
C                   FOR THE I'TH PLANT SPECIES GROUP WHEN THE N'TH
C                   CONSTITUENT IS ABSENT
C CONRAD(I)         RADIATION INTENSITY FOR MAXIMUM PHOTOSYNTHESIS OF
C                   THE I'TH PLANT SPECIES GROUP
C CONTF1(I)         PROPORTION OF OPTIMUM PHOTOSYNTHESIS ATTAINABLE AT
C                   ZERO TEMPERATURE BY THE I'TH PLANT SPECIES GROUP
C CONTF2(I)         4 NEAR COEFFICIENT IN THE TEMPERATURE DEPENDENCE
C                   FUNCTION FOR PHOTOSYNTHESIS OF THE I'TH PLANT SPECIES
C                   GROUP
C CONTES(I)         QUADRATIC COEFFICIENT IN THE TEMPERATURE DEPENDENCE
C                   FUNCTION FOR PHOTOSYNTHESIS OF THE I'TH PLANT SPECIES
C                   GROUP
C ENERGY            FACTOR CONVERTING CARBON ADDITIONS BY PHOTOSYNTHESIS
C                   TO ENERGY TERMS (KCAL. PER G.)
C EQUTL             THE RATIO OF AN ELEMENT WITHIN THE PLANT TO CARBON,
C                   WHICH WOULD BE IN EQUILIBRIUM WITH THE CURRENT
C                   EXTERNAL CONCENTRATION OF THAT ELEMENT.
C EXTTND            EXTINCTION COEFFICIENT FOR LIGHT INTERCEPTED BY
C                   ORGANIC DETRITUS (PER G. PER SQ. M.)
C EXTTNP            EXTINCTION COEFFICIENT FOR LIGHT INTERCEPTED BY
C                   PLANT MATERIAL (PER G. PER SQ. M.)
C EXTINS            EXTINCTION COEFFICIENT FOR LIGHT INTERCEPTED BY
C                   SUSPENDED INORGANIC MATERIAL (PER G. PER SQ. M.)
C EXTINW            EXTINCTION COEFFICIENT FOR LIGHT PASSING THROUGH
C                   WATER (PER M.)
C NDRIFV I          ARRAY SPECIFYING THE LIABILITYY OF THE I TH PLANT
C                   SPECIES GROUP TO DRIFT OUT OF THE SYSTEM
C PLDEP(I)          THE MEAN DEPTH IN THE WATER OF THE I'TH PLANT SPECIES
C                   GROUP SPECIES GROUP (IN M.)
C PRODFR
C PRODFF (I)        THE PROPORTION OF BIOMASS OF THE I'TH PLANT SPECIES
C                   GROUP WHICH IS DETACHED FROM THE SUBSTRATE
C RADIA             INCIDENT RADIATION REACHING THE PLANT GROUP UNDER
C                   CONSIDERATION.
C REQ(N)            TOTAL REQUIREMENT FOR THE N'TH CHEMICAL CONSTITUENT
C                   FOR PLANT GROWTH (G.PER DAY)
C RESP              CURRENT RESPIRATION RATE (PER DAY) OF THE PLANT
C                   SPECIES GROUP UNDER CONSIDERATION
C RESPC(I)          RESPIRATION RATE OF THE I'TH PLANT SPECIES GROUP AT
C                   ZERO TEMPERATURE
C RESPD(I)          LN (PROPORTIONAL INCREASE IN RESPIRATION RATE)
C                   PER DEGREE RISE IN TEMPERATURE
C SED               THE SUM OF CHEMICAL CONSTITUENTS IN SUSPENDED
C                   INORGANIC MATERIAL (USED TO CLACULATE LIGHT
```

```
C                        INTERCEPTION)
C SIEVEG(I,N)            THE RATE OF LEACHING OR EXCRETION OF THE N'TH
C                        CHEMICAL CONSTITUENT IN THE I'TH PLANT SPECIES GROUP
C                        (PER DAY)
C SPREQ(I,N)             REQUIREMENT FOR THE N'TH CHEMICAL CONSTITUENT FOR
C                        GROWTH BY THE I'TH PLANT SPECIES GROUP (G.PER DAY)
C UPCON (P,C)            UPTAKE OF THE C'TH CONSTITUENT BY THE P'TH PLANT
C                        SPECIES GROUP, PER UNIT BIOMASS CARBON, PER TIME
C                        UNIT, AND PER UNIT BY WHICH THE INTERNAL RATIO OF
C                        THIS CONSTITUENT FALLS BELOW THAT WHICH WOULD BE IN
C                        EQUILIBRIUM WITH THE EXTERIOR.
C UPCON1 (P,C)           MAXIMUM VALUE FOR RATIO OF THE C'TH CONSTITUENT TO
C                        CARBON IN THE P'TH PLANT SPECIES  GROUP
C UPCON2 (P,C)           CURVATURE FACTOR FOR RELATION OF INTERNAL TO
C                        EXTERNAL CONCENTRATION, FOR THE C:TH CONSTITUENT IN
C                        THE P:TH PLANT SPECIES GROUP.
C                        THE P'TH PLANT SPECIES GROUP.
      DIMENSION NDRIFV(20),REQ(6),SPREQ(20,6),AINT(6),AEXT(6)          0040
     1 ,IVFATE(20)
C-----------------------------------------------------------------------
C     COMMON BLOCK /ACC/ CONTAINS ACCUMULATED CHANGES, WHICH MAY BE
C     NEGATIVE.  COMMON BLOCK /ACCINC/ CONTAINS THE INCREMENTS TO THE
C     ARRAYS IN /ACC/ FOR A SINGLE TIME UNIT.
C-----------------------------------------------------------------------
      COMMON/ACC/AGAIN(3,5),EROD(3),H20(7)
      COMMON /ACCINC/ AGAINQ(3,5),ERODQQ(3), H2OQQQ(3)                 0060
C-----------------------------------------------------------------------
C     COMMON BLOCK /SPEC/ CONTAINS SPECIFICATIONS AND OTHER INFORMATION
C     COMMON TO THE WHOLE SET OF PROGRAMS, BUT EXCLUDING STATE AND
C     EXOGENOUS VARIABLES.
C-----------------------------------------------------------------------
      COMMON/SPEC/TCOVER,NCHAN,INSTRU(20),   WATER,NSPECV,NSPECA,NORGAN,  0080
     1 NFRACT,NELEM,NOLIT ,NCHECK,IDAY, ATOT, ATOTO,IYRDAY,NREPET(20)      0100
     2,NCOH(20),LISCCH(98),NCOHCU(20),NCOHOP,NFRELM,NFRAC1,NSPCOH,NDEBUG   0180
     3 ,FLOUT,MICROD, BIOMIN(98),MONTH
      COMMON/TOTALS/CVEGV(6,6),CVEGO(20,6),CVEGVO(6),AVEGV(6 ),AVEGO(20    0160
     1),AVEGVO,ABIOMA,CBIOMA(6),ALITT,CLITT(6),CORGT(6)     ,ABIOSP(20),  0180
     2TOT(6),AORGT,POPSP(20),ANIM(20,6),CBACTT(6),ABACT(3),ABACTT,         0200
     3 AVEG(20,6),ABIOM(98),ALIT(5),AORG(5),AMIN
C-----------------------------------------------------------------------
C     COMMON BLOCK /STAT/ CONTAINS THE STATE VARIABLES, AND /CHANGE/
C     THEIR INCREMENTS OR DECREMENTS FOR THE CURRENT TIME UNIT.
C-----------------------------------------------------------------------
      COMMON/STAT/ CVEG(20,6,6),CORG(5,6),POP(98),CBIOM(98,6),AQUA(6),    0220
     1 CLIT(5,6),CBACT(3,6),             DUMMY(96)
C-----------------------------------------------------------------------
C     COMMON BLOCK /METEOR/ CONTAINS THE VALUES OF EXOGENOUS VARIABLES
C     FOR THE CURRENT TIME UNIT.
C-----------------------------------------------------------------------
      COMMON/METEOR/WIRRIG,ERO,RUNSOL(6),RUNDEB(3,6), DARAIN,DAYRUN,
     1EVAP,WATTEM,DAPHOT, DAYRAD, DADUST(3,6) , EXOG(98), RAINCO(6),
     2COMPIN(6), DETIN(5,6), RUNON,FLOWIN,DRIFTV(20,6,6), DRIFTA(98,6),
     3DRIFTM(3,6), DRIFPO(98)
      COMMON /CHANGE/ CVEGQQ(20,6,6),CORGQQ(5,6),POPQQQ(98),CBIOMQ(98,6)  0250
     1,AQUAQQ(6), CLITQQ(5,6),CBACTQ(3,6)            ,DUMMQ(96)
C-----------------------------------------------------------------------
C     COMMON BLOCK/PARAM/ CONTAINS THE PARAMETERS OF THE SUBROUTINE.
```

```
C-----------------------------------------------------------------------
      COMMON/PARAM/PRODRF(20),SIEVEG(20),EXTINW,EXTINP,EXTINS,EXTIND,
     1CONRAD(20),CONTE1(20),CONTE2(20),CONTE3(20),
     2PHOT,       RESPC(20),RESPD(20),AMORT(20),DEPTH,PLDEP(20),CONNI2(        0440
     3 20,6),CONNIT(20,6),UPCON(20,6),UPCON2(20,6),ENERGY,UPCON1(20,6),
     4 DUMMYX (1659C)
C-----------------------------------------------------------------------
C      THE NAME LIST CONTAINS THE PARAMETERS THAT ARE TO BE
C      READ IN AT EXECUTION TIME.
C-----------------------------------------------------------------------
      NAMELIST/VPUT/NDRIFV,PRODRF,SIEVEG,EXTINW,EXTINP,EXTINS,EXTIND,
     * CONRAD,CONTE1,CONTE2,CONTE3,RESPC, IVFATE,
     *RESPD,AMORT,DEPTH,PLDEP,CONNI2,CONNIT,UPCON,UPCON2,ENERGY
C-----------------------------------------------------------------------
C      THE FOLLOWING SECTION DEALS WITH PHOTOSYNTHESIS.
C-----------------------------------------------------------------------
   20 FORMAT (10E11.4)                                                     0940
      NFLEM1=NELEM-1                                                       0960
      SED=0.                                                               0980
      DO 40 I=1,NFRELM                                                     1000
   40 SED=SED+CLIT(3,I)                                                    1020
      DETRI=CLIT(1,1)+CLIT(2,1)                                            1040
      A=.000001/DEPTH                                                      1060
      DO 60 K=1,NELEM1                                                     1080
   60 AEXT(K)=AQUA(K)*A                                                    1100

C..........INCIDENT RADIATION IS CUT DOWN BY SEDIMENT, SUSPENDED
C..........DETRITUS AND MUTUAL SHADING OF PLANTS AS WELL AS
C..........EXTINCTION THROUGH THE WATER TO THE MEAN DEPTH OF THE
C..........PLANT
      DO 80 K=1,NELEM1                                                     1180
   80 PEQ(K)=0.                                                            1200
      DO 300 J=1,NSPECV                                                    1220
      IF (CVEG(J,1,1).LE.0.) GO TO 300                                     1240
      IF (NELEM1.LE.1) GO TO 110
      DO 100 K=2,NELEM1                                                    1260
  100 AINT(K)=CVEG(J,1,K)/CVEG(J,1,1)                                      1280
  110 A = 0.
      DO 160 I=1,NSPECV                                                    1320
      IF (CVEG(I,1,1).LE.0.) GO TO 160                                     1340
      IF (PLDEP(I)-PLDEP(J)) 120,140,160                                   1360
  120 A=A+CVEG(I,1,1)                                                      1380
      GO TO 160                                                            1400
  140 A=A+0.5*CVEG(I,1,1)                                                  1420
  160 CONTINUE                                                             1440

C..........THE MAXIMUM ATTAINABLE PHOTOSYNTHESIS IS MODIFIED BY
C..........RADIATION, TEMPERATURE AND NUTRIENT REQUIREMENTS.
      RADIA=DAYRAD*EXP(-(EXTINP*A+((EXTINS*SED+EXTIND*DETRI)/DEPTH+        1480
     1EXTINW)*PLDEP(J)))                                                   1500
      R=RADIA/CONRAD(J)                                                    1520
      A=1.                                                                 1540
      IF (NELEM1.LE.1) GO TO 210
      DO 200 K=2,NELEM1                                                    1560
      C=CONNI2(J,K)-CONNIT(J,K)*AINT(K)                                    1580
      IF (C.LT.0.) GO TO 190                                               1600
      A=0.                                                                 1620
```

```
      GO TO 220
  180 A=A*(1.-EXP(C))
  200 CONTINUE
  210 A = B * EXP(1.-B)*A*(CONTE1(J)+CONTE2(J)*WATTEM+CONTE3(J)*
     1 WATTEM*WATTEM)
  220 RESP=RESPC(J)*EXP(RESPP(J)*WATTEM)
      SPREQ(J,1)=CVEG(J,1,1)*(DAPHOT*A-RESP)
      SPREQ(J,NELEM)=ENERGY*SPREQ(J,1)
      REQ(1)=REQ(1)+SPREQ(J,1)
      DO 280 K=2,NELEM1
      A = CVEG(J,1,1) + SPREQ(J,1)
      IF (A.LE.0.) GO TO 280
      AINTER = CVEG(J,1,K)/(CVEG(J,1,1) + SPREQ(J,1))
      EQUIL = UPCON1(J,K) * (1. - EXP(UPCON2(J,K) * AEXT(K)))
      SPREQ(J,K)=UPCON(J,K)*(CVEG(J,1,1)+SPREQ(J,1))*(EQUIL-AINTER)
      REQ(K) = REQ(K) + SPREQ(J,K)
  280 CONTINUE
  300 CONTINUE
      IF (REQ(1).LE.0) GO TO 330
      AA = (AQUA(NFRAC1) - AGAINQ(1,1))/REQ(1)
      IF (AA.GE.1.) GO TO 330
      DO 320 J = 1, NSPECV
      SPREQ(J,NELEM) = SPREQ(J,NELEM) * AA
  320 SPREQ(J,1) = SPREQ(J,1) * AA
  330 DO 350 K = 2, NELEM1
      IF (REQ(K).LE.0.) GO TO 350
      AA = AQUA(K)/REQ(K)
      IF (AA.GE.1.) GO TO 350
      DO 340 J = 1, NSPECV
  340 SPREQ(J,K) = SPREQ(J,K) * AA
  350 CONTINUE
      DO 500 J=1,NSPECV
      IF (CVEG(J,1,1).LE.0.) GO TO 500
      DO 360 K=1,NELEM
      B=AA*SPREQ(J,K)
      CVEGQQ(J,1,K)=CVEGQQ(J,1,K)+B
      IF (K.EQ.1) AGAINQ(1,1)=AGAINQ(1,1)+B
      IF (K.GT.1.AND.K.LT.NELEM) AQUAQQ(K)=AQUAQQ(K)-B
      IF (K.EQ.NELEM) AGAINQ(1,NELEM)=AGAINQ(1,NELEM)+B
  360 CONTINUE
C-----------------------------------------------------------------------
C     THE FOLLOWING SECTION DEALS WITH DRIFT.
C-----------------------------------------------------------------------
  380 FACTOR = 1. - EXP(-FLOUT*.00001/DEPTH)
      K1 = NDRIFV(J)
      GO TO (500,400,440,480     ),K1
  400 DO 420 K = 1, NELEM
      A = CVEG(J,1,K) * FACTOR
      CVEGQQ(J,1,K) = CVEGQQ(J,1,K) - A
  420 AGAINQ(2,K) = AGAINQ(2,K) - A
      GO TO 500
  440 B = FACTOR * PRODRF(J)
      DO 460 K = 1, NELEM
      A = CVEG(J,1,K) * B  -DRIFTV(J,1,K)
      CVEGQQ(J,1,K) = CVEGQQ(J,1,K) - A
  460 AGAINQ(2,K) = AGAINQ(2,K) - A
      GO TO 500
```

```
  480 CONTINUE                                                    2720
  500 CONTINUE                                                    2740
C-----------------------------------------------------------------------
C     THE FOLLOWING SECTION DEALS WITH LEACHING AND PLANT
C     MORTALITY.
C-----------------------------------------------------------------------
  520 DO 620 I=1,NSPECV                                           2840
      IF (CVEG(I,1,1).LE.0.) GO TO 620                            2860
      S = CVEG(I,1,1) * SIFVEG(I)
      CVEGQQ(I,1,1) = CVEGQQ(I,1,1) - S
      AQUAQQ(1) = AQUAQQ(1) + S
      DO600 K=1,NELEM                                             2880
      B=AMORT(I)*CVEG(I,1,K)                                      2960
      CVEGQQ(I,1,K)=CVEGQQ(I,1,K)-B                               2980
      II=IVFATE(I)
      GO TO (580,540,530,560), II
  530 CLITQQ(2,K) = CLITQQ(2,K) +B
      GO TO 600
  540 CLITQQ(1,K)=CLITQQ(1,K)+B                                   3020
      GO TO 600                                                   3040
  560 CORGQQ(1,K)=CORGQQ(1,K)+B                                   3060
      GO TO 600                                                   3080
  580 CLITQQ(2,K)=CLITQQ(2,K)+B                                   3100
  600 CONTINUE                                                    3120
  620 CONTINUE                                                    3140
C-----------------------------------------------------------------------
C     THE FOLLOWING ALLOWS FOR THE PRINTING OF DECREMENTS TO
C     THE STATE VARIABLES
C-----------------------------------------------------------------------
      IF (NDEBUG.LE.0) RETURN                                     3180
  640 FORMAT ('DCVEGQQ    ',10E11.4)                              3200
  660 FORMAT ('DCLITQQ    ',10E11.4)                              3220
  680 FORMAT ('DCORGQQ    ',10E11.4)                              3240
      WRITE (6,640) ((CVEGQQ (J,1,K),J=1,NSPECV),K=1,NELEM)       3260
      WRITE (6,660) ((CLITQQ(J,K),J=1,NOLIT),K=1,NELEM)           3280
      WRITE (6,680) ((CORGQQ(J,K),J=1,NOLIT),K=1,NELEM)           3300
      WRITE (6,700) ((AGAINQ(I,J),I=1,3),J=1,5)
  700 FORMAT ( ' AGAINQ ',10E11.4)
      RETURN                                                      3320
C-----------------------------------------------------------------------
C     THE FOLLOWING ALLOWS FOR READING OF PARAMETERS.
C-----------------------------------------------------------------------
      ENTRY VINPUT
      READ (5,VPUT)
      RETURN
      END
```

INPUT/OUTPUT EXAMPLE

A listing of a set of input cards follows, with the resulting output, using the third versions of the subroutines MEDIUM, ANIMAL and PLANT. For each input card, the number of the input statement by which it is read is indicated.

```
                                                                                         INPUT STATEMENT NUMBER
000001   000
000002   000      OFF SPRINGS  CURLEW VALLEY UTAH                                           MAIN 280
000003   000       4    9    1    1    5    3    8    3                                     MAIN 295
000004   000     150  1971  180   1    2                                                    MAIN 304
000005   000     180                                                                        MAIN 306
000006   000                                                                                MAIN 312
000007   000       3    4    3    4    4    4    3    3    3                                MAIN 314
000008   000       1    2    3                                                              MAIN 352
000009   000       1    2    4    5                                                         MAIN 374
000010   000       1    4    5                                                              MAIN 374
000011   000       1    2    6    3                                                         MAIN 374
000012   000       1    7    8    3                                                         MAIN 374
000013   000       1    7    8    3                                                         MAIN 374
000014   000       1    4    3                                                              MAIN 374
000015   000       1    4    3                                                              MAIN 374
000016   000       1    4    3                                                              MAIN 374
000017   000   PHYTOPLANKTON                BENTHIC ALGAE                                   MAIN 383
000018   000   FILAMENTOUS ALGAE            ROOTED SUBMERGED ANGIOSPERMS                    MAIN 383
000019   000   AMPHIBIA                     FISH(GILA)                                      MAIN 385
000020   000   HYALELLA                     CHIRONOMIDAE                                    MAIN 385
000021   000   ODONATA                      EPHEMEROPTERA                                   MAIN 385
000022   000   PHYSA                        ANNELIDA                                        MAIN 385
000023   000   ZOOPLANKTON                                                                  MAIN 385
000024   000   CARBON     NITROGEN    PHOSPHORUS  OTHER       ENERGY                        MAIN 393
000025   000   EGGS            LARVAE          ADULTS         YOUNG          MATURE         MAIN 394
000026   000   PUPAE           NYMPHS1         NYMPHS2                                      MAIN 394
000027   000   FINE PARTICLES  COARSE PARTICLESINORGANIC                                    MAIN 395
000028   000   PLANKTONIC      BENTHIC         ATTACHED                                     MAIN 397
000029   000   .004       .0002      .00003     .00006     .04                              MAIN 412
000030   000     .002       .0001      .00002     .00003    .02                             MAIN 412
000031   000     9.315      .4652      .06978     .13       93.15                           MAIN 412
000032   000     297.       5.94       .791       1.582     2970.                           MAIN 412
000033   000                                                                                MAIN 422
000034   000    .00003     .00009     .00013                                                MAIN 423
000035   000                                                                                MAIN 427
000036   000                                                                                MAIN 427
000037   000                                                                                MAIN 427
000038   000   100.       30.        20.        1.                                          MAIN 422
000039   000    .00003     .00009     .00013     .00025                                     MAIN 423
000040   000   .004       .00088     .000002    .000464    .040                             MAIN 427
000041   000   .004       .00088     .000002    .000464    .040                             MAIN 427
000042   000   .240       .0528      .0012      .02784     2.40                             MAIN 427
000043   000   .120       .0264      .006       .01392     1.20                             MAIN 427
000044   000    2000.     250.        30.                                                   MAIN 422
000045   000   .00003      .00009     .00013                                                MAIN 423
000046   000   .0048      .001056    .000024    .00239     .048                             MAIN 427
000047   000   .085       .0169       .000425   .041       .96                              MAIN 427
000048   000   .038       .00836     .00019     .018924    .466                             MAIN 427
000049   000    4000.     1450.      1000.       0                                          MAIN 422
000050   000   .000002    .000005    .00001     .000015                                     MAIN 423
000051   000    .0096     .0021      .000048    .0046      .096                             MAIN 427
000052   000   .055       .00121     .000275    .02739     .602                             MAIN 427
000053   000   .055       .00121     .000275    .02739     .602                             MAIN 427
000054   000   .000000000.000000000.000000000.000000000 .00000000                           MAIN 427
```

```
                                                                              INPUT STATEMENT NUMBER
000055   000   20.0       10.0       14.5       0.0                                  MAIN 422
000056   000   .000002    .000005    .000010    .00004                               MAIN 423
000057   000   .024       .00528     .00012     .011952    .240                      MAIN 427
000058   000   .09        .0198      .00045     .04482     .180                      MAIN 427
000059   000   .110       .0242      .00055     .05478     1.10                      MAIN 427
000060   000   .000000000.000000000.000000000.000000000 .00000000                    MAIN 427
000061   000    0.0       0.0        0.0        0.0                                  MAIN 422
000062   000   .0000001  .000002    .000011    .00002                                MAIN 423
000063   000   0.0        0.0        0.0        0.0        0.0                       MAIN 427
000064   000   0.0        0.0        0.0        0.0        0.0                       MAIN 427
000065   000   0.0        0.0        0.0        0.0        0.0                       MAIN 427
000066   000   0.0        0.0        0.0        0.0        0.0                       MAIN 427
000067   000                                                                         MAIN 422
000068   000   .000004    .000004    .00004                                          MAIN 423
000069   000                                                                         MAIN 427
000070   000                                                                         MAIN 427
000071   000                                                                         MAIN 427
000072   000   1000.      7550.      0.                                              MAIN 422
000073   000   .000002    .000004    .000008                                         MAIN 423
000074   000   .012       .00264     .00066     .00598     .48                       MAIN 427
000075   000    .305       .0671     .00152     .15189     3.282                     MAIN 427
000076   000                                                                         MAIN 427
000077   000    900000.   986000.    9800.                                           MAIN 422
000078   000   .0000004   .0000008   .000001                                         MAIN 423
000079   000    .00024     .00005     .000001   .00012     .0024                     MAIN 427
000080   000    .085      .0163       .00042    .0385      .784                      MAIN 427
000081   000    .0065     .00143      .00003     .00324    .07111                    MAIN 427
000082   000    .0008     .00018     .000008    .00012     .008                      MAIN 431
000083   000    .04        .0088      .0002     .006       .4                        MAIN 431
000084   000    .006      .00132     .00003      .0009      .06                      MAIN 431
000085   000    7.2       .144       .036       .02        72.                       MAIN 436
000086   000                                                                         MAIN 436
000087   000                                                                         MAIN 436
000088   000     7.2       .144      .036       .02        72.                       MAIN 438
000089   000     35.2      .704      .176       .1          352.                     MAIN 438
000090   000                                                                         MAIN 438
000091   000     3.6       1.06       .2         .4         39.384     .34           MAIN 439
000092   000     854   855                                                           MAIN 574
000093   000    FISH (GILA) CARBON                                                   MAIN 581
000094   000   GRAMS PER SQ.METER                                                    MAIN 582
000095   000   YOUNG                                                                 MAIN 585
000096   000   MATURE                                                                MAIN 585
000097   000    1474                                                                 MAIN 574
000098   000   BENTHIC HETEROTROPHIC MICRO-ORGANISMS                                 MAIN 581
000099   000   GRAMS PER SQ. METER                                                   MAIN 582
```

```
                                                          INPUT STATEMENT NUMBER
$APUT
PREFV=
8*0.,2*.1,0.,.3,7*0.,2*.3,2*0.,2*.15,0.,2*.1,70*0.,       MAIN 625 AND ANIMAL 631
4*0.,.3,.2,13*0.,2*.3,8*0.,2*.5,67*0.,                    MAIN 625 AND ANIMAL 631
0.,.5,3*0.,.1,.2,0.,2*.05,9*0.,2*.05,77*0.,               MAIN 625 AND ANIMAL 631
0.,.4,4*0.,.2,91*0.,                                      MAIN 625 AND ANIMAL 631
PREFA=                                                    MAIN 625 AND ANIMAL 631
5*0.,.02,92*0.,                                           MAIN 625 AND ANIMAL 631
6*0.,.05,91*0.,                                           MAIN 625 AND ANIMAL 631
98*0.,                                                    MAIN 625 AND ANIMAL 631
6*0.,.03,91*0.,                                           MAIN 625 AND ANIMAL 631
6*0.,.02,91*0.,                                           MAIN 625 AND ANIMAL 631
98*0.,                                                    MAIN 625 AND ANIMAL 631
98*0.,                                                    MAIN 625 AND ANIMAL 631
5*0.,.03,.1,8*0.,2*.1,81*0.,                              MAIN 625 AND ANIMAL 631
6*0.,.1,8*0.,2*.15,81*0.,                                 MAIN 625 AND ANIMAL 631
6*0.,.2,8*0.,2*.1,81*0.,                                  MAIN 625 AND ANIMAL 631
6*0.,.01,8*0.,2*.05,81*0.,                                MAIN 625 AND ANIMAL 631
5*0.,.15,.01,8*.0,2*.1,81*0.,                             MAIN 625 AND ANIMAL 631
98*0.,                                                    MAIN 625 AND ANIMAL 631
2*0.,.3,12*0.,2*.1,81*0.,                                 MAIN 625 AND ANIMAL 631
5*0.,.05,92*0.,                                           MAIN 625 AND ANIMAL 631
5*0.,.05,.01,91*0.,                                       MAIN 625 AND ANIMAL 631
5*0.,.05,.01,91*0.,                                       MAIN 625 AND ANIMAL 631
2*0.,.4,95*0.,                                            MAIN 625 AND ANIMAL 631
15*0.,2*.04,81*0.,                                        MAIN 625 AND ANIMAL 631
5*0.,.05,.01,8*0.,2*.03,81*0.,                            MAIN 625 AND ANIMAL 631
5*0.,.05,.01,8*0.,2*.03,81*0.,                            MAIN 625 AND ANIMAL 631
2*0.,.3,95*0.,                                            MAIN 625 AND ANIMAL 631
15*0.,2*.03,81*0.,                                        MAIN 625 AND ANIMAL 631
6*0.,.02,8*0.,2*03,81*0.,                                 MAIN 625 AND ANIMAL 631
6*0.,.02,8*0,2*.03,81*0.,                                 MAIN 625 AND ANIMAL 631
15*0.,2*.05,9*0.,2*1.,70*0.,                              MAIN 625 AND ANIMAL 631
15*0.,2*.08,81*0.,                                        MAIN 625 AND ANIMAL 631
15*0.,2*.02,81*0.,                                        MAIN 625 AND ANIMAL 631
4*0.,.1,.05,9*0.,2*.01,81*0.,                             MAIN 625 AND ANIMAL 631
4*0.,.2,.1,9*0.,2*.01,81*0.,                              MAIN 625 AND ANIMAL 631
4*0.,.1,.05,9*0.,2*.01,81*0.,                             MAIN 625 AND ANIMAL 631
PREFL=                                                    MAIN 625 AND ANIMAL 631
4*0.,.1,24*0.,2*.4,67*0.,                                 MAIN 625 AND ANIMAL 631
4*0.,.1,97*0.0,                                           MAIN 625 AND ANIMAL 631
PREFO=                                                    MAIN 625 AND ANIMAL 631
11*0.,.2,7*0.0,2*.1,2*0.,2*.15,0.,2*.3,70*0.,             MAIN 625 AND ANIMAL 631
8*0.,2*.4,0.,.15,7*0.,2*.05,5*0.,2*.1,70*0.,              MAIN 625 AND ANIMAL 631
PREFM=                                                    MAIN 625 AND ANIMAL 631
4*0.,2*.1,2670.,2*.1,19*0.,                               MAIN 625 AND ANIMAL 631
8*0.,2*.4,0.,.05,7*.05,2*.1,2*0.,2*.3,0.,2*.4,70*0.,      MAIN 625 AND ANIMAL 631
0.,.1,3*0.,.05,2*0.,2*.05,0.,.3,7*0.,2*.1,2*0.,2*.4,77*0.,  MAIN 625 AND ANIMAL 631
TAKE = 0.,2*.06,0.,.035,.04,.06,                          MAIN 625 AND ANIMAL 631
  2*.5,0,3*.6,0,7*.5,0,3*.7,0,2*.7,0,2*.8,0,1.,.7,        MAIN 625 AND ANIMAL 631
                      DEPTH=.75,
```

INPUT STATEMENT NUMBER

```
    NDRTFA=1,2,1,1,3*2,1,2*2,1,2*3,2*1,2*2,2*1,2*2,2*1,2*2,4*1,?1*2 ,          MAIN 625 AND ANIMAL 631
    NTRANS=2,2*1,2,3*1,3*2,12*1,7,2*1,6*2,19*0,                               MAIN 625 AND ANIMAL 631
         ASSTM=0.,2*.6,0.,3*.6,0.,2*.3,0.,3*.3,0.,3*.6,0.,3*.3,0.,             MAIN 625 AND ANIMAL 631
    2*.3,0.,2*.03,0.,2*.3,19*0. ,AMORTA=.03,2*.01,.03,3*.01,.03,2*.02,         MAIN 625 AND ANIMAL 631
    .03,3*.02,.03,3*.02,.03,3*.02,.03,2*.02,.03,2*.02,.01,2*.001,              MAIN 625 AND ANIMAL 631
19*0.,                                                                         MAIN 625 AND ANIMAL 631
TAFATE = 28*1,3*3,67*1,                                                        MAIN 625 AND ANIMAL 631
    CURVF=7*1.,7*.1,4*1.,32*.1 ,                                               MAIN 625 AND ANIMAL 631
 THRESH =50*250.,                                                              MAIN 625 AND ANIMAL 631
 AMAX = 50*300.,                                                               MAIN 625 AND ANIMAL 631
HATCON = 3*.1,4*.1,3*.005,18*.1,3*.005,                                        MAIN 625 AND ANIMAL 631
TEMCON = 50*.1,                                                                MAIN 625 AND ANIMAL 631
HATCOB = 3*.2,4*.2,3*.01,18*.2,3*.01,                                          MAIN 625 AND ANIMAL 631
THRFS2 = 98*1.,                                                                MAIN 625 AND ANIMAL 631
QONE = 98*.2,                                                                  MAIN 625 AND ANIMAL 631
    EGCOMP=98*.0004,98*.00002,98*.000003,98*.0002,98*.004,98*0. ,              MAIN 625 AND ANIMAL 631
    RCONST=50*.15 ,CONSTA=50*-1.7 ,CONSTC=50*2.7 ,CONSTB=50*160.0 ,            MAIN 625 AND ANIMAL 631
    CONSA=50*.15 ,CONSB=50*-.02 ,BOTHRE=50*0.0 ,UPTHRE=50*30.0 ,               MAIN 625 AND ANIMAL 631
    REPROD=50*.1 ,SHELP=50*.05  ,CONDAR=11*0.,2*.5,37*0.,                      MAIN 625 AND ANIMAL 631
    CONCAR=11*0.,2*1.,37*0., CONBAR = 11*0.,2*-14.5,37*0.,                     MAIN 625 AND ANIMAL 631
    CONPAR=11*0.,2*54.5,37*0. ,SIEVE=300*.005 ,                                MAIN 625 AND ANIMAL 631
    SLOPF=50*.75,CONST=50*.0002,                                               MAIN 625 AND ANIMAL 631
     FXOGEN=100.,13*0.,400.,3*0.,150.,31*0.,                                   MAIN 625 AND ANIMAL 631
          CONTA=.1 ,CONTP=.02,CONTC=.0008,CONS=3*2000.,                        MAIN 625 AND ANIMAL 631
          AMIG1=50*0. ,AMIG2=50*0. ,AMIG3=50*0.,                               MAIN 625 AND ANIMAL 631
PREFER=.2,0.,.1,2*0.,.4,0.,.45,2*0.,.4,0.,.45,8*0.,.5,4*0.,.5,18*0.,           MAIN 625 AND ANIMAL 631
ACCUM = 98*0.,                                                                 MAIN 625 AND ANIMAL 631
TSFATE = 98*1.,                                                                MAIN 625 AND ANIMAL 631
$END                                                                           MAIN 626 AND EXTERN 150
$EPUT   DRIFTV=720*0.,DRIFTA=588*0.,DRIFTM=18*0.,                              MAIN 626 AND EXTERN 150
  LIMEX=1,LIMEXO=28, $END                                                      MAIN 627 AND MEDIUM 122
$MPUT   FALL=.2,.1,.3,                         DEPTH=.75,                      MAIN 627 AND MEDIUM 122
 COAGUL =.0005 $END                                                            MAIN 628 AND VEGET  271
$VPUT     NDRIFV=2,1,2,3,16*1 ,PRODRF=20*.05 ,SIEVEC= 20*.01 ,                 MAIN 628 AND VEGET  271
          EXTINW=.2,EXTINP=.001,EXTINS=.01 ,EXTIND=.1 ,                        MAIN 628 AND VEGET  271
 CONRAD = 280.,150.,350.,280.,                                                 MAIN 628 AND VEGET  271
     CONRAD=300. , 70. ,18* 300.  ,                                            MAIN 628 AND VEGET  271
              CONTE1=20*.01,CONTE2=20*.0667 ,CONTE3=20*-.00222 ,               MAIN 628 AND VEGET  271
                                       RESPC=20*.008 ,RESPD=20*.04,            MAIN 628 AND VEGET  271
TVFATE = 20*1 ,                                                                MAIN 628 AND VEGET  271
 AMORT=4*.002              ,       DEPTH=.75 ,PLDEP=.30,.75,.1,.30,16*0.,      MAIN 628 AND VEGET  271
         CONNI2=40*2.4,80*3.2 ,CONNIT=40*80.,20*641.,60*320. ,                 MAIN 628 AND VEGET  271
     UPCON=20*1.E7,100*1.E8 ,UPCON2=20*1.,20*.2,20*.01,20*.05,40*0.,           MAIN 628 AND VEGET  271
      ENERGY=10.94,  $END                                                      MAIN 628 AND VEGET  271
```

```
                                                                                                   INPUT STATEMENT NUMBER
000001  000  36000.                                                                                  EXTERN  73
000002  000   .01         .0014      .000032    .0048     .107                                       EXTERN  75
000003  000   .01         .0014      .000032    .0048     .107                                       EXTERN  77
000004  000   .01         .0014      .000032    .0048     .107                                       EXTERN  77
000005  000   .01         .0014      .000032    .0048     .107                                       EXTERN  77
000006  000     8     9    11    12    16    17    18    19    21    47    50    51    52    61    65    74  EXTERN  81
000007  000    84    85    86    87    88   108   109   110   111   112   113   114   115   116   117   118  EXTERN  81
000008  000   119   120   121   122   123   124   125   129   130   136   137   138   145   146   147   148  EXTERN  81
000009  000   152   156   157   158   159   164   172   174   195   196   235   236   237   244   248   249  EXTERN  81
000010  000   250   263   264   280   282   283   297   299   300   311   312   313   314   319   323   325  EXTERN  81
000011  000   328   329   330   331   332   334   335   336   337   342   345   351   352   353   355   356  EXTERN  81
000012  000   364                                                                                    EXTERN  81
000013  000          .25         2.6         2.8         2.5         .25         1.50        6.4         16.  EXTERN  82
000014  000          7.9         .25         .50         3.8         .25         2.1         1.2         4.9  EXTERN  82
000015  000          1.          1.2         2.          2.1         7.1         3.8         2.2         1.   EXTERN  82
000016  000          .75        17.1         3.2        12.6         7.9         7.1         1.          3.4  EXTERN  82
000017  000          10.        27.          4.9         .5          .25         3.3         1.8         8.2  EXTERN  82
000018  000          5.25        .5          1.25        .5          1.          5.0         3.4         14.  EXTERN  82
000019  000          7.         17.          .25         7.3         1.0         23.         1.          1.   EXTERN  82
000020  000          2.          3.2         2.          .75         .5          1.25        5.0         7.5  EXTERN  82
000021  000          7.5         7.5         7.4         .75         5.0         7.1         .75         2.5  EXTERN  82
000022  000          9.0         7.5         .75         .98         .25         6.2         .25         1.25 EXTERN  82
000023  000          .25         1.0         .5          3.0         .5          .5          4.1         .25  EXTERN  82
000024  000          15.         .58         2.1         .25         3.8         1.4         2.2         1.0  EXTERN  82
000025  000          5.                                                                              EXTERN  82
000026  000   16.         1.46       .03        .06        160.                                      EXTERN  83
000027  000   .5          .5         .8         1.5        3.        4.        4.        4.            EXTERN 113
000028  000   3.          2.         1.         .5                                                   EXTERN 113
000029  000   9.24        10.31      11.46      13.16      14.29     15.12     14.59     13.59         EXTERN 114
000030  000   13.36       11.121     9.5446     19.1                                                 EXTERN 114
000031  000   148.        222.       277.       271.       322.      367.      390.      364.          EXTERN 115
000032  000   311.        276.       164.       151.                                                 EXTERN 115
000033  000   4.          4.         4.         6.         8.        10.       12.       14.           EXTERN 116
000034  000   12.         8.         6.         4.                                                   EXTERN 116
```

OFF SPRINGS CURLEW VALLEY UTAH

INITIAL REPORT ON MAY 30 1971

CONSTITUENTS OF PRIMARY PRODUCERS, G.(OR KCAL.) PER SQ.M.

	CARBON	NITROGEN	PHOSPHORUS	OTHER	ENERGY
PHYTOPLANKTON	.00400	.00020	.00003	.00006	.04000
BENTHIC ALGAE	.00200	.00010	.00002	.00003	.02000
FILAMENTOUS ALGAE	9.31500	.46520	.06978	.13000	93.15000
ROOTED SUBMERGED ANGIOSPERMS	297.00000	5.94000	.79100	1.58200	2970.00000
ALL SPECIES					
TOTAL	306.32100	6.40550	.86083	1.71200	3063.20999

CONSTITUENTS OF ANIMAL BIOMASS, G.(OR KCAL.) PER SQ.M.

	CARBON	NITROGEN	PHOSPHORUS	OTHER	ENERGY
AMPHIBIA					
EGGS	.00000	.00000	.00000	.00000	.00000
LARVAE	.00000	.00000	.00000	.00000	.00000
ADULTS	.00000	.00000	.00000	.00000	.00000
TOTAL	.00000	.00000	.00000	.00000	.00000
FISH(GILA)					
EGGS	.00400	.00088	.00000	.00046	.04000
LARVAE	.00400	.00088	.00000	.00046	.04000
YOUNG	.24000	.05280	.00120	.02784	2.40000
MATURE	.12000	.02640	.00600	.01392	1.20000
TOTAL	.36800	.08096	.00720	.04269	3.68000
HYALELLA					
EGGS	.00480	.00106	.00002	.00239	.04800
YOUNG	.08500	.01690	.00042	.04100	.96000
MATURE	.03800	.00836	.00019	.01892	.46600
TOTAL	.12780	.02632	.00064	.06231	1.47400
CHIRONOMIDAE					
EGGS	.00960	.00210	.00005	.00460	.09600
LARVAE	.05500	.00121	.00027	.02739	.60200
PUPAE	.05500	.00121	.00027	.02739	.60200
ADULTS	.00000	.00000	.00000	.00000	.00000
TOTAL	.11960	.00452	.00060	.05938	1.30000
ODONATA					
EGGS	.02400	.00528	.00012	.01195	.24000
NYMPHS1	.09000	.01980	.00045	.04482	.18000
NYMPHS2	.11000	.02420	.00055	.05478	1.10000
ADULTS	.00000	.00000	.00000	.00000	.00000
TOTAL	.22400	.04928	.00112	.11155	1.52000
EPHEMEROPTERA					
EGGS	.00000	.00000	.00000	.00000	.00000
NYMPHS1	.00000	.00000	.00000	.00000	.00000
NYMPHS2	.00000	.00000	.00000	.00000	.00000
ADULTS	.00000	.00000	.00000	.00000	.00000
TOTAL	.00000	.00000	.00000	.00000	.00000

PHYSA

EGGS	.00000	.00000	.00000	.00000	.00000
YOUNG	.00000	.00000	.00000	.00000	.00000
ADULTS	.00000	.00000	.00000	.00000	.00000
TOTAL	.00000	.00000	.00000	.00000	.00000
ANNELIDA					
EGGS	.01200	.00264	.00066	.00598	.48000
YOUNG	.30500	.06710	.00152	.15189	3.28200
ADULTS	.00000	.00000	.00000	.00000	.00000
TOTAL	.31700	.06974	.00218	.15787	3.76200
ZOOPLANKTON					
EGGS	.00024	.00005	.00000	.00012	.00240
YOUNG	.08500	.01630	.00042	.03850	.78400
ADULTS	.00650	.00143	.00003	.00324	.07111
TOTAL	.09174	.01778	.00046	.04186	.85751
TOTAL, ALL SPECIES	1.25	.25	.01	.48	12.59

ANIMAL POPULATIONS, PER SQ. M.

AMPHIBIA	
EGGS	.0000
LARVAE	.0000
ADULTS	.0000
TOTAL	.0000
FISH(GILA)	
EGGS	100.0000
LARVAE	30.0000
YOUNG	20.0000
MATURE	1.0000
TOTAL	151.0000
HYALELLA	
EGGS	2000.0000
YOUNG	250.0000
MATURE	30.0000
TOTAL	2280.0000
CHIRONOMIDAE	
EGGS	4000.0000
LARVAE	1450.0000
PUPAE	1000.0000
ADULTS	.0000
TOTAL	6450.0000
ODONATA	
EGGS	20.0000
NYMPHS1	10.0000
NYMPHS2	14.5000
ADULTS	.0000
TOTAL	44.5000
EPHEMEROPTERA	
EGGS	.0000
NYMPHS1	.0000

NYMPHS2	.0000
ADULTS	.0000
TOTAL	.0000
PHYSA	
EGGS	.0000
YOUNG	.0000
ADULTS	.0000
TOTAL	.0000
ANNELIDA	
EGGS	1000.0000
YOUNG	7550.0000
ADULTS	.0000
TOTAL	8550.0000
ZOOPLANKTON	
EGGS	900000.0000
YOUNG	986000.0000
ADULTS	98000.0000
TOTAL	1984000.0000

CONSTITUENTS OF HETEROTROPHIC MICRO-ORGANISMS, G. (OR KCAL.) PER SQ. M.

MICROBIAL TYPE	CARBON	NITROGEN	PHOSPHORUS	OTHER	ENERGY
PLANKTONIC	.00080	.00018	.00000	.00012	.00800
BENTHIC	.04000	.00880	.00020	.00600	.40000
ATTACHED	.00600	.00132	.00003	.00090	.06000
TOTAL	.04680	.01030	.00023	.00702	.46800

SUSPENDED DETRITUS CONSTITUENTS, G.(OR KCAL.) PER SQ.M.

DETRITUS TYPE	CARBON	NITROGEN	PHOSPHORUS	OTHER	ENERGY
FINE PARTICLES	7.20000	.14400	.03600	.02000	72.00000
COARSE PARTICLES	.00000	.00000	.00000	.00000	.00000
INORGANIC	.00000	.00000	.00000	.00000	.00000
TOTAL	7.20000	.14400	.03600	.02000	72.00000

BIOLOGICALLY ACTIVE SEDIMENTS, G.(OR KCAL.) PER SQ.M.

DETRITUS TYPE	CARBON	NITROGEN	PHOSPHORUS	OTHER	ENERGY	BICARBONAT CARBON
FINE PARTICLES	7.20000	.14400	.03600	.02000	72.00000	
COARSE PARTICLES	35.20000	.70400	.17600	.10000	352.00000	
INORGANIC	.00000	.00000	.00000	.00000	.00000	
TOTAL	42.40000	.84800	.21200	.12000	424.00000	
TOTAL,SEDIMENTS + DETRITUS	49.60000	.99200	.24800	.14000	496.00000	
IN WATER	3.60000	1.06000	.20000	.40000	39.38400	.340
TOTAL IN ECOSYSTEM	357.21593	7.65639	1.12126	2.33477	3572.27148	

OFF SPRINGS CURLEW VALLEY UTAH

REPORT NO. 1 ON JUNE 29 1971 (I.E., AFTER 30 DAYS OF SIMULATION)

CONSTITUENTS OF PRIMARY PRODUCERS, G.(OR KCAL.) PER SQ.M.

	CARBON	NITROGEN	PHOSPHORUS	OTHER	ENERGY
PHYTOPLANKTON	.00241	.00016	.00002	.00005	.03264
BENTHIC ALGAE	.00120	.00008	.00002	.00002	.01628
FILAMENTOUS ALGAE	6.30553	.42610	.06391	.11907	85.32008
ROOTED SUBMERGED ANGIOSPERMS	206.75732	5.59370	.74488	1.48977	2796.84775
ALL SPECIES					
TOTAL	213.06646	6.02004	.80884	1.60891	2882.21674

CONSTITUENTS OF ANIMAL BIOMASS, G.(OR KCAL.) PER SQ.M.

	CARBON	NITROGEN	PHOSPHORUS	OTHER	ENERGY
AMPHIBIA					
EGGS	.00000	.00000	.00000	.00000	.00000
LARVAE	.00000	.00000	.00000	.00000	.00000
ADULTS	.00000	.00000	.00000	.00000	.00000
TOTAL	.00000	.00000	.00000	.00000	.00000
FISH(GILA)					
EGGS	.00137	.00030	.00000	.00016	.01604
LARVAE	.00379	.00083	.00000	.00044	.04406
YOUNG	.23615	.05195	.00118	.02739	2.74287
MATURE	.07984	.01756	.00399	.01078	.92915
TOTAL	.32116	.07065	.00518	.03877	3.73211
HYALELLA					
EGGS	.00126	.00028	.00001	.00073	.01472
YOUNG	.65585	.13040	.00328	.36381	8.51858
MATURE	.01369	.00301	.00007	.00796	.19596
TOTAL	.67079	.13369	.00335	.37251	8.72926
CHIRONOMIDAE					
EGGS	.00289	.00063	.00001	.00162	.03375
LARVAE	.46552	.01024	.00233	.26650	5.85737
PUPAE	.02566	.00056	.00013	.01490	.37746
ADULTS	.00000	.00000	.00000	.00000	.00000
TOTAL	.49407	.01144	.00247	.28302	6.21858
ODONATA					
EGGS	2.88135	.14535	.02159	1.53232	30.64680
NYMPHS1	.09430	.02075	.00047	.05453	.21899
NYMPHS2	.06983	.01536	.00035	.04048	.81280
ADULTS	.00000	.00000	.00000	.00000	.00000
TOTAL	3.04547	.18146	.02241	1.62733	31.67860
EPHEMEROPTERA					
EGGS	1.10540	.05527	.00829	.58803	11.76056
NYMPHS1	.00000	.00000	.00000	.00000	.00000
NYMPHS2	.00000	.00000	.00000	.00000	.00000
ADULTS	.00000	.00000	.00000	.00000	.00000
TOTAL	1.10540	.05527	.00829	.58803	11.76056

PHYSA

EGGS	.00000	.00000	.00000	.00000	.00000
YOUNG	.00000	.00000	.00000	.00000	.00000
ADULTS	.00000	.00000	.00000	.00000	.00000
TOTAL	.00000	.00000	.00000	.00000	.00000
ANNELIDA					
EGGS	.00201	.00044	.00011	.00118	.09447
YOUNG	.22912	.05041	.00114	.13271	2.86752
ADULTS	.00000	.00000	.00000	.00000	.00000
TOTAL	.23114	.05085	.00125	.13388	2.96198
ZOOPLANKTON					
EGGS	.00015	.00003	.00000	.00009	.00177
YOUNG	.11818	.02266	.00059	.06207	1.26387
ADULTS	.00540	.00119	.00003	.00313	.06872
TOTAL	.12373	.02388	.00062	.06528	1.33437
TOTAL, ALL SPECIES	5.99	.53	.04	3.11	66.42

ANIMAL POPULATIONS, PER SQ. M.

AMPHIBIA	
EGGS	.0000
LARVAE	.0000
ADULTS	.0000
TOTAL	.0000
FISH(GILA)	
EGGS	40.1007
LARVAE	22.1910
YOUNG	14.7940
MATURE	.7397
TOTAL	77.8254
HYALELLA	
EGGS	802.0143
YOUNG	136.3711
MATURE	16.3645
TOTAL	954.7499
CHIRONOMIDAE	
EGGS	1604.0285
LARVAE	790.9524
PUPAE	545.4844
ADULTS	.0000
TOTAL	2940.4653
ODONATA	
EGGS	7994.5922
NYMPHS1	5.4548
NYMPHS2	7.9095
ADULTS	.0000
TOTAL	8007.9565
EPHEMEROPTERA	
EGGS	2994.9645
NYMPHS1	.0000

NYMPHS2	.0000
ADULTS	.0000
TOTAL	2994.9645

PHYSA

EGGS	.0000
YOUNG	.0000
ADULTS	.0000
TOTAL	.0000

ANNELIDA

EGGS	401.0071
YOUNG	4118.4073
ADULTS	.0000
TOTAL	4519.4144

ZOOPLANKTON

EGGS	665730.4375
YOUNG	956845.0625
ADULTS	95102.2471
TOTAL	1717677.7344

CONSTITUENTS OF HETEROTROPHIC MICRO-ORGANISMS, G. (OR KCAL.) PER SQ. M.

MICROBIAL TYPE	CARBON	NITROGEN	PHOSPHORUS	OTHER	ENERGY
PLANKTONIC	.07439	.01637	.00011	.01116	.74392
BENTHIC	.28589	.06290	.00143	.04288	2.85890
ATTACHED	.05888	.01295	.00029	.00883	.58885
TOTAL	.41917	.09222	.00184	.06288	4.19167

SUSPENDED DETRITUS CONSTITUENTS, G.(OR KCAL.) PER SQ.M.

DETRITUS TYPE	CARBON	NITROGEN	PHOSPHORUS	OTHER	ENERGY
FINE PARTICLES	1.93381	.28582	.00883	.97372	21.82839
COARSE PARTICLES	4.53483	.11442	.01572	.02993	56.04709
INORGANIC	.00000	.00000	.00000	.00000	.00000
TOTAL	6.46864	.40024	.02455	1.00365	77.87548

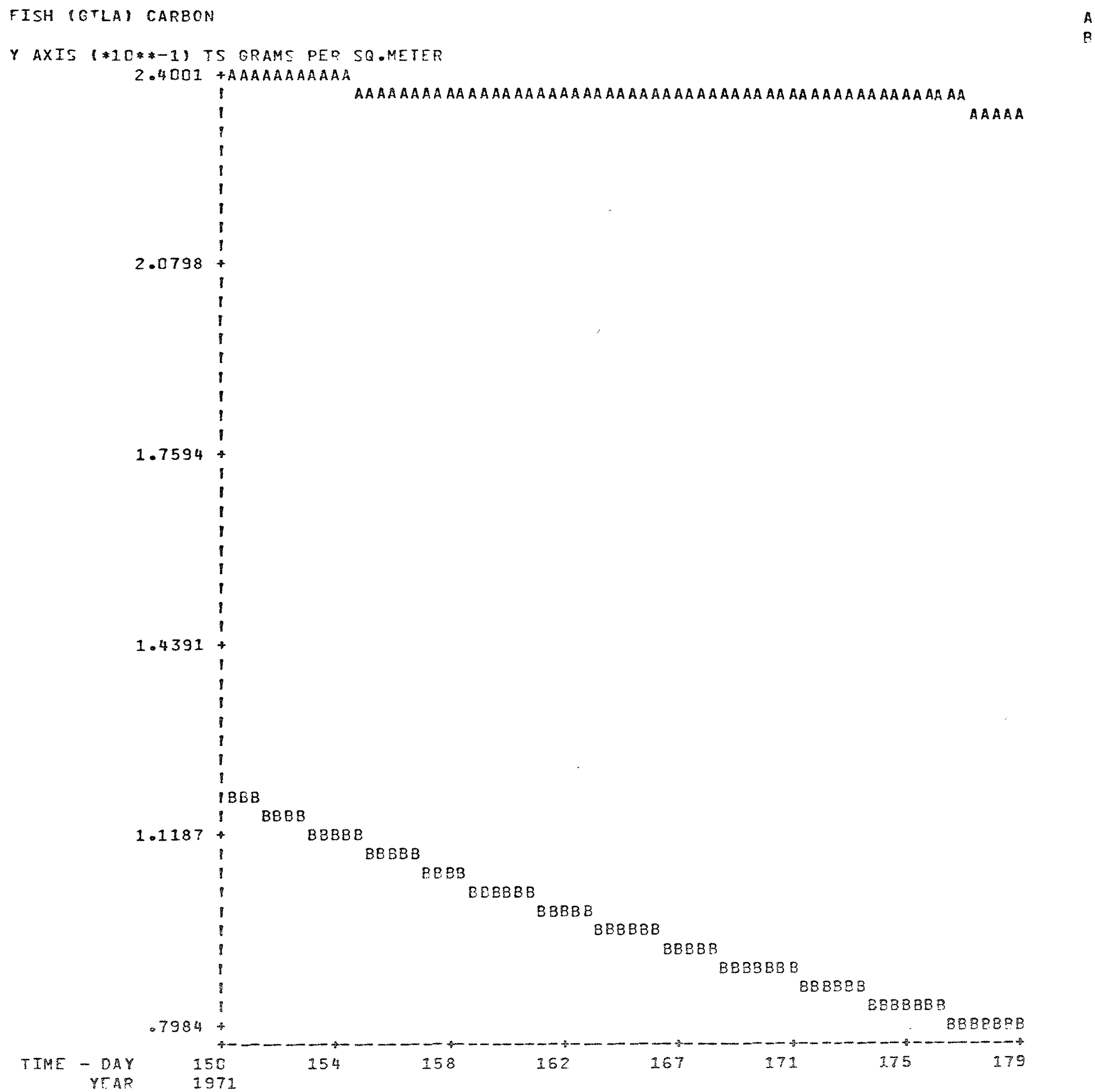
BIOLOGICALLY ACTIVE SEDIMENTS, G.(OR KCAL.) PER SQ.M.

DETRITUS TYPE	CARBON	NITROGEN	PHOSPHORUS	OTHER	ENERGY	
FINE PARTICLES	17.70561	1.23997	.08644	2.97023	182.19699	
COARSE PARTICLES	39.84143	.80852	.18050	1.44292	410.43488	
INORGANIC	.00000	.00000	.00000	.00000	.00000	
TOTAL	57.54704	2.04849	.26694	4.41314	592.63187	
TOTAL,SEDIMENTS + DETRITUS	64.01568	2.44873	.29149	5.41679	670.50735	
						BICARBONATE CARBON
IN WATER	80.96652	1.11116	.20543	.49155	40.02017	.5300
TOTAL IN ECOSYSTEM	283.49306	9.08823	1.14574	10.19740	3623.33115	

ACCUMULATED NET GAIN OR LOSS TO ECOSYSTEM	WATER	INERT PARTICLES	CARBON	NITROGEN	PHOSPHORUS (

TO OR FROM ATMOSPHERE	.00000	.00000	6.60000	.33000	.04950
BY RUN-OFF OR RUN-ON	.00000	.00000	.00000	.00000	.00000
TOTAL	.00000	.00000	6.60000	.33000	.04950

ACCUMULATED PRECIPITATION = .0 MM. - THAT IS, .0 G. PER SQ.M.


FISH (GTLA) CARBON
A
B
Y AXIS (*10**-1) TS GRAMS PER SQ.METER
2.4001
2.0798
1.7594
1.4391
1.1187
.7984
TIME - DAY
YEAR
150
1971
154
158
162
167
171
175
179

BENTHIC HETEROTROPHIC MICRO-ORGANISMS

Y AXIS (*10**-1) TS GRAMS PER SQ. METER